1994 / MÁRCIUS ÁRA: 279 FT

ÚJ ALAPLAP

SZÁMÍTÁSTECHNIKAI MAGAZIN MÁGNESLEMEZ MELLÉKLETTEL

Levelezés elektronikusan

Feltérképezett élettér

Készítsünk képeskönyvet

Kivel van az erő?



A HÓNAP TÉMÁJA:

SZÖVEG, AZ VAN...

CD-lemez a magazinban

Hírháló — először, de nem utoljára

Interaktív függvényvizsgálat

Dinamikus programozás — élőben

A MÁGNESLEMEZEN:

Mintalevelek E-mailhez
Karakterkonverzió dBase szövegben
IDE winchesterek azonosítása
A véletlen Ctrl+Alt+Del
Játék: Csillagkapu

ÚJ ALAPLAP

A Mikroszámítógép Magazin és az Alaplap hagyományait folytató számítástechnikai folyóirat

Megjelenik havonta, mágneslemez melléklettel

Főszerkesztő:
Faklen Pál

Főszerkesztő-helyettes:
Varga János

Szerkesztők:
Jakab Ágnes
Sziebig Andrea

A szerkesztőbizottság tagjai:
Barna László, Broczkó Péter, Brüll Károly, Csórián Sándor, Farkas Ernő, Feleki Zoltán, Fridl György, Herczeg József, Lóth Tamás, Sík Zoltán, Vargha Dénes, Vékony Tamás, Villányi László, Zoltai Péter

Szerkesztőség és kiadó:
1538 Budapest I., Márvány u. 17.
Telefon: 156-3211/200, 214
Fax (manuális): 156-3211/201

Terjesztés:
Héber Sándor

Hirdetésszervezés:
Árvai Katalin

Külföldi hirdetések:
Publicitas
Nemzetközi Médiaügynökség
1537 Budapest I., Márvány u. 17.
Telefon: 156-1182 Fax: 175-3539

Felelős kiadó:
Faklen Pál

Nyomtatás:
Zalai Nyomda, Zalaegerszeg
Felelős vezető: Somogyi Tibor

Terjeszti:
A Magyar Posta, a Nemzeti Hírlapkereskedelmi Rt, a Hírker Rt, az Extra-Hír, számos számítástechnikai szaküzlet és más terjesztő

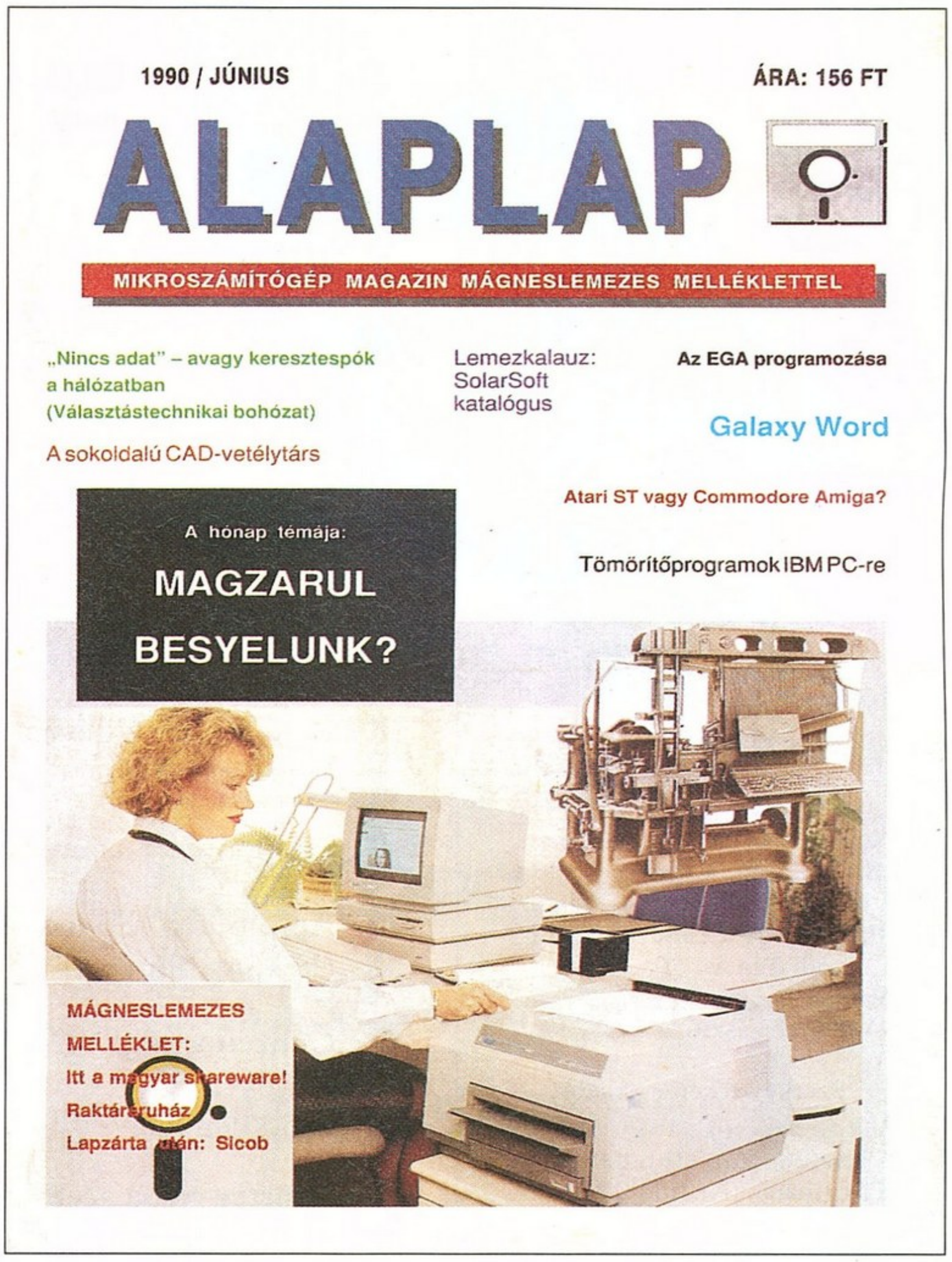
Előfizethető a kiadónál:
Új Alaplap Kiadói Kft, 1538 Budapest, Pf. 571
Átutalás: Agrobank 219-93789

Példányonkénti ár: 279 Ft
Évi előfizetési díj: 2 820 Ft

Külföldre terjeszti a Kultúra, H-1389 Budapest, Pf. 149

HU ISSN 1217-7598

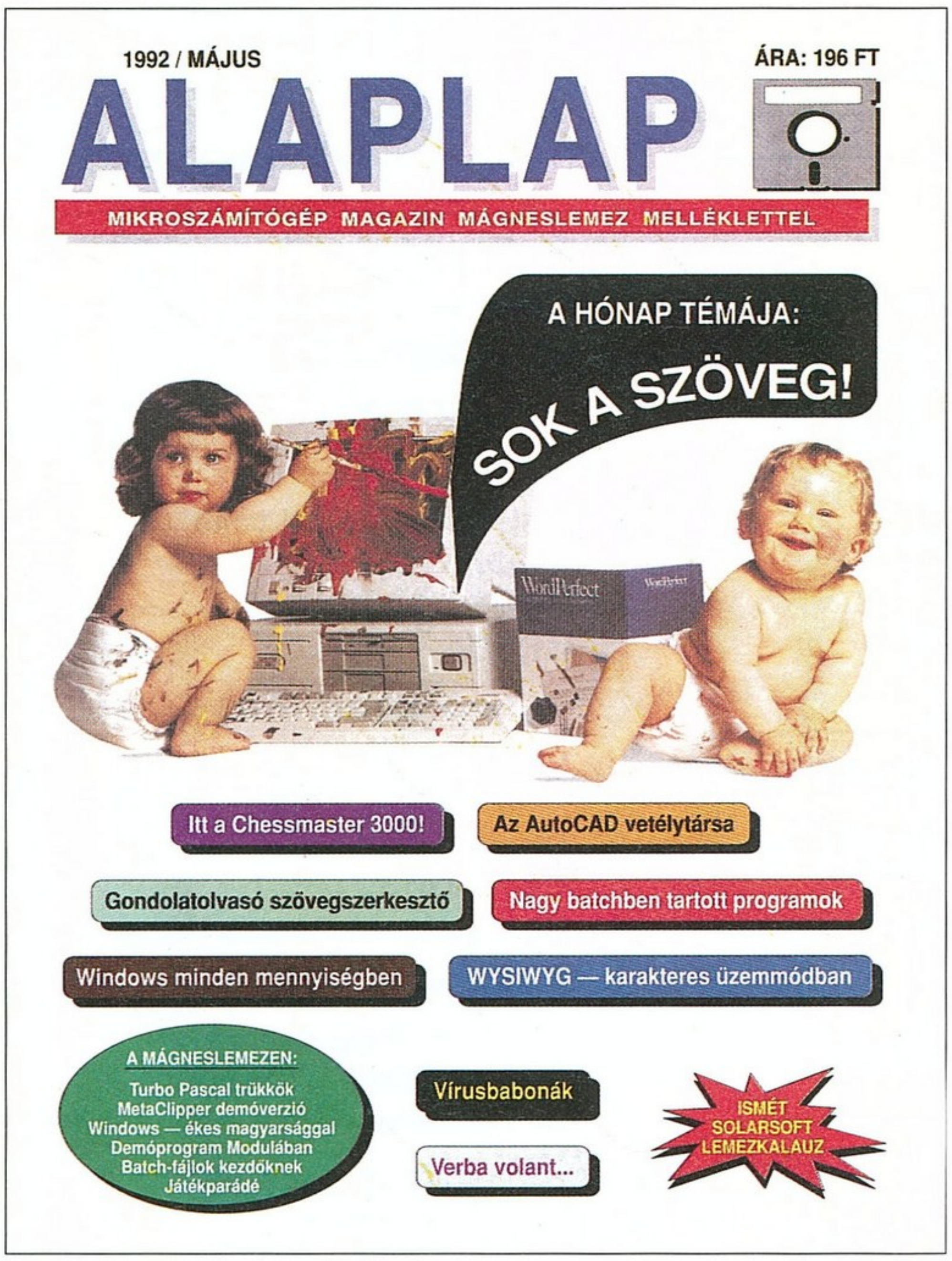
Már megint? Még mindig!



Amikor 4 évvel ezelőtt a Mikroszámítógép Magazin „átvedlett" Alaplappá, nem kellett hosszasan töprengenünk, hogy az első számhoz mit válasszunk a hónap témájául. Felhalmozódott problémák tömege tette indokolttá a szövegszerkesztés napirendre tűzését. Sajnos örökzöld témára leltünk! Kevés kivételtől eltekintve minden akkori — és későbbi hasonló összeállításunkban közölt — cikk változtatás nélkül újra kinyomtatva is teljesen aktuális lenne. Ez bizony azt jelenti, hogy a számítástechnika legelterjedtebb használati módjáról Magyarországon kiállított bizonyítvány továbbra is igen gyengécske.

A számítógépek 1990 óta sokkal korszerűbbek és olcsóbbak lettek, a grafikus felületek a szövegszerkesztéshez minden szépet és jót kínálnak, a helyesírásellenőrzők és más segédprogramok sorra születnek és tökéletesednek... csak éppen a szövegek nem akarnak a lehetőségekhez igazodni.

Mondhatnák erre a bigott számítástechnikusok, hogy itt is a HU-szindrómáról van szó. A HU rövidítés az ő szóhasználatuk szerint nem a Hungarian megfelelője, hanem egy „magyangol" terminus technicusé (hülye user). Egy percig sem akarjuk mentegetni a felhasználói slendriánságot vagy tudatlanságot, de végignézve a hibás szövegeken, az is látszik ám, hogy a gépek tudorai sokszor elfelejtenek néhány dolgot megemlíteni (vagy beállítani) azoknak, akiket az írógéptől átültettek a számítógép klaviatúrája elé. Így készülnek az elvben több szűrőn áthaladó, mégis hibáktól hemzsegő nyomtatványok, hirdetések, üzleti dokumentumok — korrekt magyar szöveg helyett „kalapos betűkkel", rossz elválasztással, elemi helyesírási hibákkal...



Aki számítógépet használ, az kivétel nélkül szöveget is ír vele, és majdnem kivétel nélkül szerkesztget. Ahányan vagyunk, annyiféleképpen tesszük ezt, mert eltérő a tudásunk, műveltségünk, nyelvérzékünk, fogalmazóképességünk, igényességünk, szépérzékünk... Bár nagyon különböző szinteken, de tulajdonképpen mindannyiunknak vannak nyelvi fogyatékosságai. A kérdés csak az, hogy hajlandók vagyunk-e meghaladni tegnapi önmagunkat, és egy kicsit jobban tisztelni a nyelvet, a szöveget. Az igazi tét sokkal nagyobb, mint első pillanatra látszik. A szöveg ugyanis maga a gondolat, a mondanivaló, az üzenet.

S ha a fentiek után valaki esetleg azt hiszi, hogy összeállításunkban most sopánkodni fogunk egy kiadósat „szövegeink állapotán", az téved. Éppen mert korábbi eszmefuttatásaink (és kesergéseink) megtartották időszerűségüket, s mert olvasótáborunk nagyobbik része több mint 4 éve kitart mellettünk, nem bocsátkozunk ismétlésekbe, hanem igyekszünk új nézőpontokból megközelíteni a régi témát: érzékeltetni a határvonalat a szövegszerkesztés és a kiadványszerkesztés között, gondolatokat ébreszteni az adatbáziskezelőkben lévő szövegekről, sőt bepillantani a Unix szövegkezelésének első látásra talán bonyadalmas világába is.

Az I. és a II. kódháború

Szöveg-előtörténet

A szövegszerkesztés a számítástechnika kezdeti időszakában inkább csak a programsorok bevitelének megkönnyítését szolgálta.
Az igencsak körülményesen kezelhető akkori szövegszerkesztők helyett a nagyobb tömegű adat bevitelére eleve más adathordozókat (lyukkártya, lyukszalag stb.) használtak.
Később azonban a géppel folytatott párbeszéd egyre jobban elszakadt a számok nyelvétől, és uralkodóvá vált a szöveg.

A legelső szövegbeviteli eljárás a ma is meglévő copy con parancs őse volt a CP/M operációs rendszer alatt. A pip paranccsal egyaránt lehetett beírni szöveget vagy programokat. Hátránya „mindössze" annyi volt, hogy csak az éppen beírt sorban, még annak elküldése előtt lehetett javítani. Ha hibáztunk, akkor mindent újra kellett írni. Ezt a problémát azután megoldotta a másik, máig kísértő őskövület, az Edlin. A programozást tanulókat sokáig ezzel az editorral kínozták a tanárok. E primitív szüleményt a DOS még mindig hordozza magával, más kérdés, hogy néhány mazochista programozón kívül aligha használja bárki is, tehát szidni sem érdemes.

Nem véletlen, hogy a fapados megoldások nem elégítették ki a programozókat, és valami használhatóbbat kellett kitalálni az akkoriban elterjedt nagygépekhez. Először még kissé bumfordi volt ugyan, de azután egyre csiszolódott a VMS operációs rendszerre készült Xedit, amelynek DOS-os átiratát már Kedit néven tiszteli a PC-s világ. Mindenesetre az első, valóban jól használható editorrá fejlődött a DOS alatt. (Az Új Alaplap szerkesztői is évek óta főleg Kedittel dolgoznak, csak alkalmanként használnak egyéb szövegszerkesztőket, amikor például konvertálni kell a nem 8-bites ASCII-ban kapott kéziratokat.)

Valójában a full screen editoroktól számíthatjuk a szövegfeldolgozás kezdeteit. A teljes képernyős editor abban különbözik az Edlin soreditortól, hogy a kurzorral az egész képernyőn mozoghatunk, szerkeszthetünk, sort törölhetünk és szúrhatunk be, anélkül, hogy bonyolultabb parancsokat kellene alkalmazni, vagy újra kellene írni a szöveget.

A PC-s világban innen ágazott el két irányba a szövegszerkesztők fejlődése. Egy részük továbbra is megmaradt az ún. tiszta ASCII formátum határain belül — azaz szövegállományukat bármivel el lehetett olvasni, mert a „kocsivissza-soremelés" és a „fájl vége" speciális karaktereken kívül más segédjeleket nem használtak. Tipográfiai megoldásokat sem lehetett alkalmazni (leszámítva az esztétikailag egyébként is kétes értékű sorkizárást), hiszen a DOS nem tudott volna a formázó utasításokkal mit kezdeni. A Kediten kívül ebbe a szövegszerkesztő kategóriába tartozik például az IBM belső szoftvere, a Personal Editor, vagy a kereskedelmi programok közül a Norton Editor, amelyek korai verziói a mai napig is szinte szabadszoftverként forognak a programozók körében, és olyannyira megszokott és bevált eszközök, hogy sem azok továbbfejlesztett változatai, sem a későbbi programozói editorok nem tudták teljesen kiszorítani őket.

A szövegszerkesztők fejlődésének másik irányát az időközben eléggé háttérbe szorult WordStar jelölte ki. Ők hozták létre az ún. vegyes ASCII formátumot. Az ilyen szövegállományokban a szövegkarakterek és a tipográfiai parancskódok békésen összekeverednek a formátumparancsokkal. Ennek viszont az a hátránya, hogy csak azzal a programmal (sőt néha csak azzal a verzióval) lehet a szövegeket elolvasni és kinyomtatni, amelyikkel létrehozták azokat.

A kétféle szövegszerkesztési koncepció közül a szöveget a nyomdai megjelenítési lehetőségekhez egyre inkább közelítő második megoldás vált népszerűbbé. A számítástechnika kihívást intézett a nyomdák hagyományos szedési technológiája ellen, és ezt a csatát elég rövid idő alatt meg is nyerte. A letölthető karakterkészletek, és az egyre nagyobb felbontású printerek elterjedésével a formai gazdagság megvalósítására képes szövegszerkesztő programok fokozatosan a formakészítési műveleteket is áthelyezték a nyomdákból az irodákba. Innen vezetett az út egyenesen a DTP teljes eszköztárának kifejlődéséhez.

Magyarországon az igényes szövegfeldolgozás mindennapos tevékenységgé válását főleg a magyar nyelv ékezetes betűi gátolták. A professzionális minőségű, ékezethelyes szöveg előállítását az új fényszedési technológiájú nyomdai szedőrendszerek elég gyorsan tudták produkálni, a számítógép azonban sokáig csökönyösen ragaszkodott az ékezettelen angol, illetve a bizonyos ékezeteket már tartalmazó német vagy francia betűkészletekhez. Az IBM annak idején valahogy „megfeledkezett" más nyelvek ékezetes betűinek a kódrendszerbe történő integrálásáról, s ennek mind a mai napig isszuk a levét.

Szerencsére több itthoni cég is felismerte: meg kell oldani a magyar nyelv speciális karaktereinek

— bevitelét a billentyűzeten,
— ábrázolását a képernyőn,
— kinyomtatását a printereken.

A dolgot bonyolította, hogy akkor még nem terjedtek el a letölthető grafikus karakterkészleteket használni tudó EGA- és VGA-kártyák. A magyar karakterek használatához ezért módosítani kellett — az embargó miatt természetesen minden leírás és dokumentum nélkül — a Hercules- és CGA-kártyák karaktergenerátorát, valamint a printerekét is. Ahány „ház"-, annyi formátum jött létre. Mivel az adatbázisok történetileg ebben a vegyes környezetben képződtek, a szabványos megoldások hiánya ma is sokakat sújt, a konvertálás nincs mindenütt megoldva.

A formátumok kavalkádja már kezdetben is nagy káoszt eredményezett.

Más kódkiosztást használt az Állami Biztosító, mást a Belügyminisztérium, mást a HM, hogy a szűkebb körben született egyéni megoldások tucatjairól ne is beszéljünk. Ekkor jött az SZKI, amely szintén kidolgozott egy kódtáblát. Ennek logikája tiszteletre méltó, mert érintetlenül hagyta a nyugat-európai nyelvek speciális szövegbetűit, s helyette az amerikai kódtábla felső régióiban tobzódó keretező karaktereket „lopta el". Az átdefiniálással létrejött az ún. Ventura-kódtábla. Nyomdai célokra ez nagyszerűen bevált, mert a nyomdák más eljárással kereteznek, mint a programozók. Ezzel szemben a Ventura-kód a már megírt karakteres rajzolatú programokat vagy szövegeket a képernyőn igencsak elrondítja, nézzünk csak rá e kódkészlet betöltése után akár a Norton Commander keretvonalaira, akár egy szövegben a szervezeti felépítést bemutató ábrára.

Az iménti esztétikai és funkcionális dilemma áthidalására a CWI-Számítástechnika szerkesztősége más alapelvre épülő kódtáblát javasolt a magyar ékezetes betűkre. Abból indultak ki, hogy vannak a kibővített nemzetközi kódtáblának főleg a latin nyelvekben előforduló speciális karakterei, amelyek Magyarországon (legalábbis programozói berkekben) alig használatosak, és ha már a kompatibilitást le kell szűkíteni, inkább a német és angolszász nyelvterületet válasszuk. Ők tehát nem a keretező vonalak, hanem a hasonló alakú betűk helyére definiálták a nemzetközi kódtáblából hiányzó hosszú ékezetes magyar magánhangzókat:

Å	ì	ò	ô	º	ù	û	ÿ
Á	Í	Ó	ő	Ő	Ú	ű	Ű

A ránézésre is valamelyest hasonló betűkép előnye, hogy így — egy kis tréning után — a nem átalakított mátrixprintereken kinyomtatott szöveg is viszonylag jól olvasható és érthető, bár korrektül magyarnak egyáltalán nem nevezhető. (A Ventura-kód hasonló esetben még az edzett szemnek is állandóan rejtvényfejtés.) A CWI-kódokon lévő ékezetes magyar betűknek az eredetiekkel fennálló hasonlósága viszont bumerángként is működött, mert nagyon sok helyen megelégedtek a hasonlósággal, nem gondoskodtak a megfelelő megoldásról (letölthető nyomtatóvezérlés, EPROM-csere, betűkészlet stb.), így ma is vidáman használják a „kalapos betűket". S ami még szomorúbb: lézerprinteren, a legújabb Windows programokkal készült friss szövegekben ugyanúgy ott „éktelenkednek" ezek a majdnem magyar betűk. (Számlákon, szerződések szövegében, „készen" leadott hirdetésekben, reklámokban stb.)

Ilyen előzmények után kezdődött el (a Microsoft és az IBM jóvoltából) a II. kódháború. Az történt, hogy a „kódtáblán kívüliségük" miatt folyton elégedetlen kelet-európai országok összes renitens ékezetes betűjét összezárták egyetlen „kód-gettóba", a 852-es kódtáblába. Ráadásul elvárták, hogy ettől mi most nagyon boldogok legyünk, mert lám-lám, ők biz' gondolnak ránk is. Sajnos a 852-es nem a fejlett országokhoz való felzárkózásunkat szolgáló eszköz, hanem inkább a „vak vezet világtalant" elvre emlékeztető mankó.

Az előzőekkel nem kompatibilis 852-es kódtábla használatának erőltetése, megtoldva a mindennapi életet és a szakma véleményét negligáló magyar klaviatúraszabvánnyal, ismét megbolygatta a konvenciók alapján már kialakulóban lévő informatikai konszenzust. Még nem látszik a II. kódháború kimenetele, de az már igen, hogy mi lehet a folytatás: a III. kódháború, amikor is a „nagyoknak" olyan kedvük támad, hogy bevezetik a Unicode mamutprogramot, amelyben a kínai, arab vagy grúz íráselemek mellett természetesen megtalálhatók a magyar ékezetes betűk és írásjelek is. Csak abban — mondhatni naná! — néhány karakter ismét nem azon a kódszámon szerepel, mint az előző kódtáblákban.

„Eszi, nem eszi, nem kap mást!"

Kis János

Írógép vagy nyomda?

A címbeli kérdést a számítógépes környezetre kell értelmezni, pontosabban arra, hogy hol végződik a szövegszerkesztés, és hol kezdődik a kiadványszerkesztés. Jogosan tehetjük fel ezt a kérdést, mert számos szövegszerkesztőt időközben DTP-rendszerré fejlesztettek, és a már eleve DTP-nek készült programcsomagokban is ott lehet a „word processing" képessége.

A kétféle funkció elkülönítése még akkor sem felesleges, ha a határvonalakat nehéz meghúzni, mert lényegében az egyik a tartalmat, a másik a formát szolgálja, és jó, ha tudjuk, nekünk éppen mire van szükségünk. A szakirodalmi adatok alapján megpróbáltuk tehát összeírni azokat a műveleteket, amelyek a mezsgye kijelöléséhez támpontokat nyújthatnak.

Szövegszerkesztő

Kötelező:

— Szöveg bevitele (billentyűzetről, fájlból)
— Utólagos szerkesztés (törlés, beszúrás, blokkműveletek)
— Sorigazítás (jobbra, balra, középre, szélekre zárva)
— Táblázatszedés (tabulálás)
— Karakterhelyes nyomtatás (önmagában vagy segédprogrammal)

Szabadon választott:

— Helyesírás-ellenőrzés
— Elválasztás
— Képletszedés
— Három alapvető betűtípus (normál, kövér, kurzív)
— Léniák, grafikai elemek alkalmazása
— Képek beillesztése és kinyomtatása
— Részmentési, beszúrási, átirányítási műveletek

DTP

Kötelező:

— Szövegelrendezés hasábokra és keretekbe
— Sorok tetszőleges tagolása függőlegesen és vízszintesen
— Sokféle betűfokozat, betűtípus és betűkészlet kezelése
— Képletszedés és táblázatszerkesztés
— Tipográfiai elemek (léniák, keretek, tónusok stb.) alkalmazása
— Grafikai elemek (iniciálék, rajzok) beillesztése, körülszedése
— WYSIWYG, azaz nyomtatáshű megjelenítés a képernyőn is
— Képfeldolgozási, képszerkesztési, képmanipulálási lehetőség
— Tartalomjegyzék és indexek generálása
— Automatikus lap- és fejezetszámozás
— Homogén színek használata és színkivonatának előkészítése

Szabadon választott:

— Beépített szövegszerkesztő
— Helyesírás-ellenőrzés
— Betűképek torzítása, alakítása
— Teljes színbontás előkészítése (4 szín nyomásához)
— Más programrendszerek adatainak direkt átvétele

Szóról szóra — minden szinten

Keressünk egy jó szövegszerkesztőt!

Nyolc éve még a Commodore 64 volt legtöbbünk vágyainak netovábbja. Nemcsak a hozzáférhető (többségében játék)programok sokasága miatt, de könnyű kezelhetőségének és szövegszerkesztési lehetőségének köszönhetően is. Az EasyScript (az egyik legelterjedtebb C64-es szövegszerkesztő) sok olyan szolgáltatással rendelkezett, melyek némelyike még a mai PC-s utódok eszköztárából is hiányzik. A gépek számával együtt a szövegszerkesztő programok is szaporodtak, s ma már az elvégzendő feladatokhoz akár 10-20 vagy több, közel egyenértékű program közül választhatunk.

A nekünk megfelelő program megtalálásához tisztában kell lennünk az elvégzendő feladatokkal és az egyéb követelményekkel is. Semmiképpen nem elég az önreklámozó programismertetők alapján választani. Érdemes sorra venni az ellenőrizendő dolgokat:

— Akik majd használják a szövegszerkesztőt, azok mennyire szoktak már hozzá a PC-hez.

— Hardverpark, felszereltség (PC, nyomtató, hálózat, egyebek).

— A korábban használt szövegszerkesztők, a meglévő anyagok szövegformátuma.

— A beírandó szövegek mennyisége és egy-egy állomány tipikus mérete.

— A szövegek formázásával szembeni igény: csak a beírás, az adatrögzítés-e a lényeg, vagy kiadványszerű szövegeket kell előállítani.

— A kész anyagok kibocsátási módja (papíron, faxon, lemezen fájlban, hálózaton vagy egyéb módon).

— A vásárlás és a folyamatos költségek fedezete.

A költségek szándékosan kerültek az iménti lista végére. Bármennyire is ezek befolyásolják majd végleges döntésünket, nem szabad az elemzést eleve ehhez láncolva elkezdeni. Fontos, hogy előbb sorra vegyük a valóban szakmai szempontokat, és csak utána keressük meg a szükséges kompromisszumot a kívánatos és az anyagilag lehetséges között. (Ez persze szinte minden racionális beszerzési döntésre igaz! Mint az is, hogy nem mindig a legolcsóbb a legelőnyösebb.)

Az emberi tényező

A számítástechnikusok időnként megfeledkeznek arról, hogy akik a gépek többségét használják (ügyviteli, pénzügyi, jelentésgyártási és más feladatokra), azok számára a számítógép fekete doboz, és semmi ingerük nincs arra, hogy belenézzenek, többet megtudjanak róla, mint ami napi munkájukhoz minimálisan szükséges. (Gondoljanak csak arra, hogy a számítógépen dolgozók többsége nő, viszont az Alaplapról 1993-ban készített felmérés szerint a lap olvasóinak 95 százaléka férfi!)

Ebből fakadóan a szövegszerkesztési munka sebességére és hatékonyságára is komoly hatással van a felhasználó „előélete". Ha az új program megjelenése és kezelése nem tér el gyökeresen attól, amit valaki előzőleg már megszokott, akkor könnyebben fogadja el, és hatékonyabban tudja használni. Ezért is jók azok a kezelői felületek, amelyeket személyre lehet igazítani, mert így a program alkalmazkodik a kialakult szokásokhoz, a billentyűkombinációk használatának esetleg több év alatt begyakorolt mozdulataihoz.

Feladattípusok

Szövegszerkesztésről beszélve nem szerencsés egy kalap alá venni az adatrögzítés jellegű mechanikus szövegbevitelt a sokrétűbb szövegszerkesztési és szövegformázási munkával. Fontos tudni, hogy egy szövegszerkesztőnek melyik a domináló felhasználási módja. Felesleges (és költséges) magunkat „túlmotorizálni", kiadványszerkesztési képességet invesztálni oda, ahol azt utána soha nem fogják használni.

A szövegszerkesztő programok egy része elsősorban a programozók számára készült. Számukra nem a szövegformázási szolgáltatások bősége, hanem a fejlesztő környezettel való szoros kapcsolat, a „program programozhatósága" a legfontosabb. A Norton Editor, a MultiEdit és társai éppen ezért formai szempontból nagyon egyszerűek, kevés memóriát igényelnek, gyorsak és egyszerűek. Akárcsak végtermékük, a tiszta ASCII szövegfájl. Megjelenítésük sem igényel grafikus felületet.

Más azok helyzete, akiknél minden megírt anyagnak „valamilyen pofát kell adni". Vállalatok és intézmények különböző részlegeinél a titkárságok ontják magukból a jelentéseket, körleveleket, elszámolásokat, terveket, címkéket... Nekik olyan szövegszerkesztő kell, amely ha nem is nyomdai, de legalább igényes irodai kivitelű szövegeket tud kinyomtatni, mégpedig viszonylag egyszerűen, és rögtön a beírást követően. Ehhez már többféle betűkészlet, formázósablon és sok egyéb kell, ami megtalálható például a Word, a WinWord, a WordPerfect, az AmiPro programokban. Aki nagyon igényes, az ábrákkal, képekkel, táblázatokkal is meg tudja tűzdelni az anyagot, de itt már a külalak szempontjából nem mindegy, hogy mártixnyomtatót, lézerprintert vagy egy tintasugaras nyomtatót használunk.

Akik a felokosított (és monstrumokká bővített) szövegszerkesztők produktumainál is igényesebb — például nyomdai sokszorosításra szánt — anyagokat akarnak készíteni, azok tapasztalataink szerint azzal a kombinációval érhetik el a legjobb eredményt, hogy az egyszerű, formázatlan szövegek készítésére rendezkednek be, **akárhol és akárhány helyen,** és a szövegeket hozzáértő munkatársak formázzák meg a megfelelő tudású kiadványszerkesztő rendszereken — **egy vagy néhány helyen.**

A szerzőktől ma már a legtöbb folyóirat szerkesztősége (az Új Alaplap is) formázatlan ASCII szövegfájlban

kéri a kéziratot. Ennek előállításához nem kell sem WinWord, sem WordPerfect, elég a Norton Editor, a Kedit vagy akár a DOS Edit programja. A szerkesztés tartalmi része, az általában túl sok redundanciát tartalmazó szövegek megrövidítése, a stiláris és nyelvtani hibák kijavítása, a helyesírás ellenőrzése a szerkesztőségben egy hasonlóan „primitív" szövegszerkesztő rendszeren történik, mert ez a legegyszerűbb, leggyorsabb, leghatékonyabb. Az anyag tördeléséhez, azaz a kész oldalak kialakításához, tipográfiai megkomponálásához viszont már valódi DTP-rendszer szükséges, és olyan valaki, aki rendelkezik bizonyos nyomdai ismeretekkel, grafikai képességekkel, esztétikai érzékkel.

A fenti kombinált megoldás több előnnyel is jár. Szétválik a mechanikus adatrögzítési munka és a komolyabb szakértelmet igénylő szerkesztés, ami megtakarítást jelent munkabérben és eszközparkban, mert kiválóan megfelel a célnak egy kiszolgált XT vagy egy 286-os AT, akár 20 megás winchesterrel is.

Do you sprichst po mad'arsky?

A hazai fejlesztésű szövegszerkesztők mellett sorra készülnek a sikeres külföldi (főleg amerikai) programok magyar nyelvű változatai. Ez üdvözölendő jelenség, mert hát a számítógépet használók nagy részének a PC talán az angol nyelv miatt is idegen közeg, és a magyar vagy magyarított programok esetleg mégis csak közelebb kerülhetnek hozzájuk.

E megoldás is hordoz azonban problémákat. Például azt, hogy nem mindig követhető a frissítés, vagy hogy a kapcsolódó, nem magyarított programok angol kifejezéseit a magyarított menüpontokkal egyeztetni néha nehezebb, mint két érthetetlen, de „bötűről bötűre" egyformán írt idegen szót.

A magyarítással könnyen elvész az informatikai kompatibilitás. A németek a honosításban jóval előttünk járnak, de nekik is igen sok gondot okoz a menüszerkezet és a makrónyelv németesítése, vagy a függvénynevek németre „ferdítése" a táblázatkezelő programokban. S ha figyelembe vesszük, hogy az eddig megjelent kézikönyvek is mindig az angol változatokat, az angol menüket ismertetik, akkor már nem is látszik olyan ésszerűnek a magyarított programok használata. (Gerő Judit és Reich Gábor remek WinWord-könyvének a magyar programverziót bemutató változata üdítő kivétel.)

1. táblázat

Komplett szerkesztési konfigurációk költségtételei

1. Négy beíró, két szerkesztő állomás, összekötés nélkül:

286/1MB/40MB/mono VGA/265K/1,2MB	4 x 50 000
386/8MB/200MB/SVGA/1MB/1,2+1,44 MB/egér	2 x 110 000
CD-ROM + streamer (120/250 MB-os)	2 x 50 000
Mátrixnyomtató (A3-as, 9 tűs)	1 x 35 000
Lézernyomtató	1 x 80 000
Szkenner (A/4-es, 400-as felbontás)	1 x 60 000
MS Word 6.0 for DOS	4 x 36 000
MS Word 6.0 for Windows	2 x 36 000
Windows 3.1	2 x 13 000
CorelDraw 4.0 CD-n	1 x 49 000
Encyclopedia of Clipart (CD)	1 x 5 000
800 db magyar TrueType font	1 x 15 000
QuarkXpress for Windows	1 x 78 000

2. Négy beíró, két szerkesztő állomás, hálózatba kötve:

286/1MB/40MB/mono VGA/265K/1,2MB	4 x 50 000
386/8MB/200MB/SVGA/1MB/1,2+1,44 MB/egér	2 x 110 000
CD-ROM + streamer (120/250 MB-os)	2 x 50 000
Ethernet-kártya + kábel	6 x 6 500
Lantastic 5.0	6 x 13 000
Mátrixnyomtató (A3-as, 9 tűs)	1 x 35 000
Lézernyomtató	1 x 80 000
Szkenner (A/4-es, 400-as felbontás)	1 x 60 000
MS Word 6.0 for DOS	4 x 36 000
MS Word 6.0 for Windows	2 x 36 000
Windows 3.1	2 x 13 000
CorelDraw 4.0 CD-n	1 x 49 000
Encyclopedia of Clipart (CD)	1 x 5 000
800 db magyar TrueType font	1 x 15 000
QuarkXpress for Windows	1 x 78 000

3. Munkacsoport hálózatba kötve:

386/4MB/120MB/mono VGA/512K/1,2+1,44 MB/egér	4 x 90 000
486/8MB/340MB/SVGA/1MB/1,2+1,44 MB/egér	2 x 160 000
CD-ROM + streamer (120/250 MB-os)	2 x 50 000
Ethernet-kártya + kábel	6 x 6 500
Lantastic 5.0	6 x 13 000
Mátrixnyomtató (A3-as, 9 tűs)	1 x 35 000
Lézernyomtató	1 x 80 000
Szkenner (A/4-es, 400-as felbontás)	1 x 60 000
MS Word 6.0 for Windows	6 x 36 000
Windows 3.1	6 x 13 000
CorelDraw 4.0 CD-n	1 x 49 000
Encyclopedia of Clipart (CD)	1 x 5 000
800 magyar TrueType font	1 x 15 000
QuarkXpress for Windows	1 x 78 000
Modem (9600-os MNP5, adat/faxmodem)	1 x 39 000

Integrált csomagoknál, így a windowsos szövegszerkesztőknél különösen zavaró lehet, ha az összedolgozó programoknak csak egyik része magyar nyelvű, a többi angol. Sajnos a Microsoft a magyarításnál ragaszkodik a 852-es kódkészlethez, ami eltorzítja a más kódokkal készült anyagok megjelenítését. Nemes gesztus a visegrádi négyek együttműködését a kódkészlettel is támogatni, de mi a magunk részéről az angol nyelvű programokkal való kompatibilitás érdekében szívesen lemondanánk róla.

Szövegtovábbítás

Ahol szöveg készül, ott azt tárolni és továbbítani is kell. Ahogy régen a papírhegyek nőttek, úgy most az elektronikusan tárolt információk „gigásodnak" meg „terásodnak".

Fokozatosan a szövegtovábbítás nagy része is átterelődik az elektronikus országútra, faxon, LAN- vagy WAN-hálózaton, modemen keresztül. Ha a fájlok továbbításából ki akarjuk iktatni a közvetítő adattárolót (a lemezt, a streamert), akkor a szövegszerkesztőt kiegészítő további programokra (LAN szoftver, kommunikációs programok, E-mail) van szükség.

A listavezető Windows-alapú rendszerek (WinWord, WordPerfect for Windows, AmiPro stb.) eleve felkínálnak ilyen szolgáltatásokat, és a hálózati szoftvereket is elláthatjuk a szükséges kiegészítésekkel. A fentiekhez csatlakozik egy Magyarországon még újnak számító fogalom, a csoportmunka (workgroup). Itt nemcsak arról van szó, hogy közös kiszolgálóról (serverről) indítunk el programokat, és közösen használjuk a hálózati nyomtatót, a CD-t, a modemet stb., hanem egy készülőben lévő anyag szerkesztésekor mód van a munkakópiát körbeküldeni az érintetteknek, hogy olvassák át, s tegyék meg észrevételeiket, vagy szükség esetén írják bele a szükséges javításokat, kiegészítéseket.

Mi az olcsó, mi a drága?

Tételezzük fel azt a ma még nem tipikus esetet, hogy minden géphez megvesszük az ott szükséges szövegszerkesztő, hálózati, kommunikációs stb. programokat. A DOS-os programok (legalábbis a Microsoft termékeinél) semmivel nem, vagy csak alig olcsóbbak, ellenben a futtatásukhoz szükséges hardveren jócskán spórolhatunk. Egy 6 munkahelyes egység például az 1. táblázatban látható módokon alakítható ki. Természetesen a bemutatott példák mellett számos egyéb megoldás is választható, a nagyságrend mindenesetre érzékelhető.

A CD-olvasó hasznossága ma már nem szorul magyarázatra, hiszen a lézerlemezen szállított TrueType fontok, a szövegek díszítésére használható sok ezernyi grafika, a CD-ről telepíthető programok sokasága és a CD-ROM adatbázisok kimeríthetetlen tárháza önmagáért beszél.

A streamer egészen más megfontolásból szerepel a listában. A szerkesztési munka során óriási mennyiségű archiválandó anyag keletkezik. Ezek biztonságos és gazdaságos tárolását legjobban streamer kazettákon lehet megoldani. A 2. táblázat összehasonlító adataiból kiderül, hogy 120 Mbájtnyi nem tömöríthető anyag archiválása a különböző adathordozókon mennyibe kerül. Kommentár felesleges.

2. táblázat
120 MB adatmennyiség tárolási költségei

Floppylemez (100 db 1,2 MB = 120 MB)	6 000
Floppylemez (90 db 1,44 MB = 129 MB)	9 000
Floptical lemez (6 db 21 MB = 126 MB)	12 000
Bernoulli-disk (105 MB)	13 000
MO-lemez (128 MB)	9 000
Merevlemez (120 MB)	22 000
Streamer kazetta (120 MB)	3 000

Végül, milyen lehetőségeik vannak azoknak, akik otthonra, a családi számítógépre akarnak szövegszerkesztő programot telepíteni? A nagy tudású programok, a majdnem kiadványszerkesztők az átlagháztartás számára szinte megfizethetetlenek. Maradnak az olcsóbb programok, a freeware, public domain, shareware szoftverek. Még mindig feltételezve, hogy jogtisztán beszerzett programokkal akarjuk merevlemezünket feltölteni, elmondható, hogy az olcsó programokból is széles a választék. Több helyen is kínálnak 2000 és 10 000 forint közötti áron jó minőségű szövegszerkesztőket. Magyar nyelvű, hazai fejlesztésű programok is vannak közöttük.

A leggazdagabb választék azonban a shareware programoké. Bármelyik hazai shareware könyvtár (SolarSoft, PannonSoft, Keszo, BBS-ek stb.) katalógusában 30-40 olyan programot is találunk, maximum 500 forintos lemezáron, amelyek közül valamelyik bizonyosan megfelel igényeinknek.

Tudásban, sebességben ezek a programok semmivel sem maradnak el az átlagos kereskedelmi szoftverek mögött. Akinek modemje is van, az még ezt az 500 forintot is megspórolhatja, ha valamelyik BBS-ről hívja le a csomagot.

Van tehát mivel, megtanulhatjuk, hogy hogyan, így csak azon múlik minden, hogy legyen mit — lehetőleg értelmes szöveget — szerkeszteni.

Nagy Gábor

Gyorsítás és tömörítés — haladóknak

A windowsos szövegszerkesztők mindegyike fejlett makrózási és formázási szolgáltatásokkal rendelkezik. Adunk ehhez még egy hasznos tippet azoknak, akik rendelkeznek valamelyik röptömörítő programmal (DoubleSpace, Stacker, SuperStore, Xtradrive stb.).

Egy 120 Mbájtos merevlemezen a szabad területen hozzanak létre két 40 Mbájtos tömörített fájlt. Az így előállított, kezdetben 80 Mbájtos D: meghajtóra érdemes a Windowst és a választott szövegszerkesztőt telepíteni, magyar ékezetes TrueType fontokkal kiegészítve. A szintén tömörített E: meghajtót hagyjuk meg a dokumentumoknak és a mintafájloknak.

A fenti megoldás több előnnyel is jár. Mivel a Windows és a szövegszerkesztő csak az .INI fájlokat módosítja a D: meghajtón, a program nem fog lassulni a programfájlok töredezettsége miatt, sőt gyorsulás érzékelhető, ha a gép processzora elég nagy teljesítményű (386DX/33-40). Az E: meghajtó a folyamatos munka következtében időnként defragmentálásra (az egybe tartozó állományok összefüggő lemezterületre való átrendezésére) szorul. Így a lemezoptimalizáló programnak csak kisebb területtel kell foglalkoznia, hiszen a Windows és a szövegszerkesztő a másik tömörített fájlban van, és gyorsabb lesz a rendbetétel. Emellett a rendelkezésre álló szabad területet is pontosabban tudjuk megbecsülni. Lévén csak egyféle adattípus a lemezen, a becsült tömörítési arány, amellyel a tömörítő a szabad területet felméri, közelebb fog állni a valóságoshoz.

N. G.

WP vagy DTP

Befellegzett Gutenbergnek?

A fényszedés megjelenése a DTP irányába megtett egyik döntő lépésnek tekinthető. Az első fényszedőrendszerek azonban drágák és bonyolultak voltak, ráadásul a már akkor elterjedt számítógépekkel sem mutattak semmiféle kompatibilitást.
A tudatos elszigetelési törekvésnek alig 6 évvel ezelőtt egy másik lapnál e cikk szerzője (a Új Alaplap főszerkesztőjével együtt) közvetlenül is szenvedő alanya volt. Az a magyar szedőrendszer, amelynek fejlesztésében felhasználóként kénytelenek voltak „részt venni", nemcsak messze elmaradt a számítástechnikai fejlődéstől, hanem az „alkotók" azt akarták velünk elfogadtatni, hogy számítógépen előzőleg már rögzített szövegeket is újra be kelljen kopogni a szedőrendszerbe.
(Lásd kompatibilitás és felhasználóbarát rendszerek!)

A korai levilágítók a katódsugárcsővel előállított képet betűnként fényképezték nyomdai filmre. Emellett egy másik rendszer is élt, amely betűképeket fényképezett hasonló módon egy mintafilmről (masterről) a szövegfilmre. Mindkét megoldás lassúnak, nehézkesnek bizonyult. Ugyanakkor az egyre jobban terjedő elektronikus képfeldolgozás újabb eszközöket kívánt a filmen történő rögzítéshez. Kapóra jött, hogy a finommechanika és a lézertechnika akkor vált elérhető áron a polgári fejlesztések számára is hozzáférhetővé.

Lapleíró eszperantó

Viszonylag gyorsan megjelentek a Linotype, Agfa Compugraphic nyomdai levilágítók, amelyek a saját raszter-image processzor (RIP) segítségével képpontsoronként fényképezték rá a filmre a képet, a szöveget, a rasztert — egyetlen műveletben. Ekkor jött és győzött az Adobe, amely korábban a Xerox cégnél kifejlesztett egységes lapábrázolási nyelv teljes szerzői gárdájának átvételével és az eredeti koncepció továbbfejlesztésével megalkotta a PostScript lapleíró nyelvet, s az minden komoly lézerlevilágító és lézernyomtató eszperantójává vált, megteremtve a közös nevezőt a különböző gyártmányú szedőrendszerek között.

Az ún. I. magyarországi kódháború idején a számítógépeken már vígan dolgoztak az első CWI-kódos WordStar programokkal, amelyek némi tipográfiát is kölcsönöztek az anyagoknak. A vállalatok kezdték leépíteni IBM composeres szedőkapacitásukat. Ugyanakkor megvolt az igény, hogy a számítógépen igényesebb kivitelű kiadványokat is lehessen készíteni.

A nagy ugrást az SZKI-féle Xerox Ventura első, majd második változatának, és az XY-Write magyarul megtanított szövegszerkesztő programnak a megjelenése jelentette. Az SZKI másolásvédelemmel és magas áron hozta forgalomba a Venturát, ami — utólag rekonstruálva a történteket — meggyorsította a szoftver elterjedését, mert a bitvadászok „kénytelenek voltak" a védelmet feltörni, és ezzel a programot széles körben munkaeszközzé tették. Ez a „védtelen" Ventura annyira népszerű lett, hogy kényelmi okokból általában még azok is ezt használták, akik a jogtiszta változatot megvették. A „kiadványszerkesztés" szinonimája hamarosan a „venturázás" lett. Az SZKI-kódkiosztás pedig eközben az igényes nyomdai felhasználók körében is elterjedt.

A HP lézernyomtatók, később a PostScript lézerprinterek megjelenésével tehát megszakadt a hagyományos elektronikus szedés térhódítása, az ólomszedésen alapuló nyomdai technológiák pedig a kihalás szélére sodródtak.

A Quarkot sem kell védeni

Még alig vakarták le a másolásvédelmet a régi Venturáról, amikor megjelent a program nem védett Windows-verziója, s az egykori Titán Kisszövetkezet kidolgozott hozzá egy racionális Windows-kódkészletet, amellyel a CWI nyersszöveget betöltve (minden konverzió nélkül) ékezethelyes kép keletkezett. A Ventura Windows-verziója azonban inkább volt poloskatenger, mintsem használható program, így — legalábbis külföldön — a PageMaker jött, látott és tarolt. A Windows-alapú DTP programok esetében ez lett a szabvány. A jelenlegi befutónak azonban az eredetileg Macintoshra készült QuarkXpress jól sikerült Windows-átirata ígérkezik. Persze egy kis kerülővel, mert a magyar piac továbbra sem vevő a másolásvédett programokra, így a Quark esetében is inkább az eredeti amerikai szoftvert részesíti előnyben, a hazai környezetnek megfelelő segédprogramokkal felszerelve.

Kenyérféltő nyomdászok

A szövegszerkesztő (word processor) és a kiadványszerkesztő (desktop publishing) közötti határvonalat egyre nehezebb meghúzni. Az AmiPro és a WinWord sokoldalúságban, grafikai, tipográfiai lehetőségekben versenyez a Ventura for Windows-zal. Ugyanakkor a „született" DTP programok is igyekeznek magukról elhitetni, hogy mindazokat a funkciókat tudják, amelyekkel egy jó szövegszerkesztő rendelkezik. Ez a kölcsönös mimikri persze nem oldja fel azt az alapvető ellentmondást, amely a kétféle tevékenység eltérő jellegéből fakad. A szövegszerkesztő alapfunkciója a kézirat *tartalmi* kialakítása, és közvetlen rokonságát a karakteres manipulálás, az adatbáziskezelés, a táblázatkezelés környékén kell keresni, míg a DTP alapfunkciója a kézirat *formai* megjelenítése, ezáltal legszorosabban a grafikus felületekhez, a rajzoló, festő és illusztráló programokhoz kötődik.

A szövegszerkesztés és a DTP kimeneti eszközeiben sincs nagy különbség. Egyaránt használják a professzionális

PostScript levilágítót vagy 9 tűs Epson printert. (Ez utóbbi minősége professzionálisnak persze korántsem nevezhető...) A jelenlegi szövegszerkesztők egy része képes nyomdai levilágítók „keze alá" is dolgozni. Van is rá igény, mert egy kisebb példányszámú szakkönyv megjelentetésének legjobb módja, hogy a szerző megírja (például Wordben), azt egy korrektor elolvassa és kijavítja (először papíron, vagy akár rögtön képernyőn), majd a programhoz (és valamelyest lehetőleg a tipográfiához is) értő valaki formát ad neki. Az eredmény ugyan rendszerint nem éppen nyomdai remekmű, de használható, elfogadható külalakú termék. Még a nyomdász szakma sokáig bevehetetlen váraként tisztelt matematikai képletszedésre is kénytelenek voltak kitűzni a fehér zászlót: a Winword 6.0, valamint a WordPerfect for Windows kifogástalan képletszerkesztési lehetőségeket kínál a „laikusok" számára is.

Annál jogosabb és megalapozottabb viszont a nyomdászok másik kritikája, hogy a DTP megöli a kiadványok lelkét, esztétikumát, egyéniségét, nyelvhelyességét. Az uniformizálás, az előre gyártott (és rosszul alkalmazott) panelek valóban veszélyforrások, és bizony sok olyan ember is gépközelbe kerül, akinek sem tipográfiai érzéke, sem ízlése, sem grafikai képessége nincs, a magyar helyesírással és stílussal pedig hadilábon áll, de ettől még zavartalanul „kiadványszerkeszt", grafikonokat rajzol, hirdetéseket készít, hírleveleket gyárt... Ez ellen egyelőre nem sokat lehet tenni. Talán a szaktanfolyamok adhatnának többet a kiadványkészítés elektronikus eszközökre átdolgozott szaktudásából. Persze még ebben az esetben is maradna egy rendíthetetlenül magabiztos „kemény mag", amely sem autodidakta tanulással, sem tanfolyamokon való részvétellel nem hajlandó elismerni, hogy a kiadványkészítés ugyanolyan bonyolult és összetett szakma, mint a számítógépes programozás.

A jelek szerint a nagyon igényes nyomdai feladatokon kívül nem sok munka őrizhető meg Gutenberg avatott mestereinek privilégiumaként. Az ólombetű kora elmúlt, de azért továbbra is körülvesz bennünket a nyomtatással előállított információk sokasága. Csak éppen a minőséggel van baj. Persze a Gutenberg-korszak múltjára visszapillantva azt is mondhatja valaki, hogy a DTP gyermekbetegségeinek sem lesz nyoma ötszáz év múlva... De tessék mondani, nem lehetne ezt a határidőt egy kicsit előbbre hozni!?

Kis János

Epsilon — egy kis különbség

Programozói szinten

Vajon írnak-e még programokat? Vagy talán már mindegyiket csak rajzolják? Én azt hiszem, hogy írnak még programot, hiszen a rengeteg szoftvert valakinek el kell készítenie. Ha pedig így van, akkor szükség van programozói editorra. Az Epsilon, a Lugaru cég terméke sok éve van a piacon, mostani változata a 6.5-ös verziószámot viseli.

Az angol terminológia meg is különbözteti a programmer's editor és a wordprocessor fogalmát. A programozók különösen „háklisak" arra, hogy milyen eszközöket használnak — próbáljon csak valaki egy Brief-hívőt vagy egy Kedit-rajongót a másik táborba átcsábítani! Minden programozónak megvan a maga kedvenc editora. Amelyikről most írok, az viszonylag kevéssé ismert Magyarországon, de van néhány olyan képessége, amely miatt igencsak megéri megnézni, kipróbálni.

A Laguri eredendően törekedett arra, hogy a termék a Unix-felhasználók által jól ismert Emacs minél hűbb DOS-os mása legyen. Ez pedig igen fontos, hiszen az Emacs a Unix-világban kváziszabványnak számít, és nem is ok nélkül: példátlanul sokrétű, gazdag szolgáltatásaival és végtelen konfigurálhatóságával valósággal világ a világban. Ráadásul a Unix egyre elterjedtebb nálunk is, egyre több programozónak kell mindkét platformon dolgoznia. Ezért az Epsilon ismerete igencsak hasznos lehet.

Epsilon-sztenderd

Manapság egy editornál vannak amolyan kötelező funkciók — ezeket mindig elmondják, de ezek nélkül nem is jelenik már meg a piacon termék. A szokásos funkciók természetesen mind megtalálhatóak az Epsilonban. Lehet egyszerre több fájlt szerkeszteni, vagy egy fájl több részét, mindezt sok, változatos elhelyezkedésű ablakban. A beolvasott fájlok méretét csak a rendelkezésre álló memória korlátozza. (Ebbe természetesen a konvencionális mellett az EMS, XMS, UMB is beleértendő.) Annak, hogy a fájlokat a program teljes egészében a memóriában kezeli, számos következménye van. Nem lehet akármekkora egy fájl, de a műveletek persze így sokkal gyorsabbak. Az Epsilont sem arra szánták, hogy sok-sok Mbájtos állományokat kezeljen.

Ki lehet jelölni sorokból álló vagy téglalap alakú blokkokat, ezek — mint várható — másolhatók, törölhetők, sokszorozhatók stb. Rengeteg fájlon (azaz inkább pufferen) belüli mozgási lehetőség van. Az érdekesebbek a mondat, paragrafus, C-függvényenkénti mozgás. Megjelölhetünk helyeket, amelyekhez aztán egy gombnyomással vissza lehet térni. Ezeknek a helyeknek nevet is lehet adni, és a „visszaugrásnál", amikor ki akarjuk választani, hogy hova is ugrunk, a helyek nevét és a hozzájuk tartozó sort is látjuk egy ablakban.

Az undo és a redo (visszacsinálás és vissza-visszacsinálás) mértékét csak a memória korlátozza. Ráadásul megkülönböztetődik a pufferben való mozgás és a fájl megváltoztatása. Tehát vissza lehet sétálni korábbi pozíciókra és állapotokra, de ugyanezt külön is kérhetjük.

Epsilonnyi többlet

Tetszőleges billentyűkiosztást definiálhatunk magunknak — ahogy ez elvárható —, sőt: a szerkesztett fájl típusa (kiterjesztése) alapján is más és más lehet egyes billentyűk hatása. Az Epsilonnak saját kiterjesztő nyelve van, ez az EEL, a C-hez igen hasonló szintaxissal. Ami az egyik érdekesség, és itt már kezdődnek a specialitások, hogy magát a futó programot is jórészt ebben a kiterjesztő nyelvben írták. További öröm, hogy a teljes forrás rendelkezésre áll, így tanulni is lehet, de buzgó és

ügyes programozók teljesen átírhatják a rendszert. Az EEL-ben írt programok, rutinok menet közben be- és kitölthetőek, sőt még egy igen jól használható debugger is segít az esetleges hibák megtalálásában.

A billentyűk kezelése is gazdagabb az editoroknál megszokotthoz képest. Nemcsak a Ctrl+betű és Alt+betű kombinációkhoz rendelhetünk parancsokat, de Ctrl+Alt+betű összetételek is képezhetők. Így még logikusabbá tehetjük a parancskiosztást, ha van hozzá türelmünk. Az Epsilon értelmesen alkalmazkodik a menet közbeni változásokhoz, amikor helpet kérünk, mindig az aktuális billentyűket mutatja a parancsoknál.

Ritka és kellemes többlet a fájlok, pufferek különbségének megmutatása. Ha az egyik ablakban egy programnak az egyik, a másikban pedig a másik változata van, akkor kérhetjük, hogy egy harmadik ablakban jelenjenek meg a két fájl közötti különbségek, vagy interaktív módon végigmehetünk az eltéréseken. Szintén rendkívül hasznos a grep egy integrált változata, amellyel a pufferek vagy a lemezen lévő fájlok közül az adott keresési feltételnek megfelelőket választhatjuk ki. Egyébként az is a felhasználó kényelmét szolgálja, hogy fájlnév vagy parancs megadásánál működik az ún. completion (teljessé tétel). Elég az elejét beírni valaminek, és a Tab gomb hatására az Epsilon befejezi a gépelést. Ha több lehetséges teljessé tétel van, akkor ezeket egy ablakban kínálja fel.

Epsilon — C-s szemszögből

A programokat írók, különösen a C-programozók kényelmét szolgálja, hogy alkalmazkodik a szintaxishoz, és változtatható stílusban be- és kiugrasztja a sorokat, megtalálja a kezdő és végző zárójeleket. Ezenkívül a „TAGS" parancsra végignézi a megadott fájlokat, és egyfajta adatbázist készít a deklarációkból. Aztán csak egy gombnyomás, és ha egy modulban egy függvényhivatkozásnál elvesztünk, már ott is az a fájl, amelyben a függvényt definiáltuk, és mindjárt szerkeszthetjük is. Ezek az információk ráadásul projekttől függők lehetnek. Ha akarjuk, kilépéskor az Epsilon elmenti az adott szerkesztési állapotot, legközelebb ott folytathatjuk (fájlok, ablakbeosztás, pozíciók), ahol abbahagytuk.

A keresés és változtatás könnyű, de rendkívül gazdag lehetőségeket kínál. Van inkrementális keresés, szelektív változtatás, mindez előre, hátra vagy globálisan, és eközben reguláris kifejezésekkel a legbonyolultabb képződmények is megtalálhatók. Például azok a változók, amelyek harmadik betűje nagybetű, és a sor végén vannak. Próbálja csak valaki ezt másképp megkeresni!

Epsilon alatt

Az egér sokszor jól jön az editorban, ha nagyot akarunk ugrani, ha blokkot jelölünk ki, stb. De az egérkurzor idegesítő, amikor gépelünk, és a görgetősávok sok drága helyet foglalnak el. Az Epsilon szellemesen oldja ezt meg. Az egér „alszik" gépeléskor, de feléled, ha megmozdítjuk, és a képernyő széle csak akkor válik görgetősávvá, ha odahúztuk az egeret. Ideális kompromisszum!

Végül a legérdekesebb, hogy programot futtathatunk az Epsilon alól. De nem ám úgy, hogy kimegyünk a DOS-ba, hanem az Epsilon is aktív marad, és közben az outputot egy később szerkeszthető ablakba gyűjti. Persze nem minden programmal működik ez a megoldás, csak a nagyon jól viselkedőkkel, és a memória sem valami sok ilyenkor (340 kbájt körüli). De ez még így is olyan hasznos dolog lehet, amelyet érdemes megnézni, és szükség esetén használni.

A kézikönyv világos, de a kiterjesztő nyelvről szóló rész bizony lehetne alaposabb. A cég rendkívül korrekt, telefonon, faxon vagy BBS-en minden segítséget megad, az árak mérsékeltek. A teljes verzió 250 dollár körüli, a rendszeresnek mondható új verzióké 35-40 dollár. Összefoglalva az Epsilon nagyjából az ideális programozói editor, de nem kezdőknek való.

Horlai János

Apró pofon a Windowsnak!

Bevalljuk, nem vagyunk híján a jóleső érzéseknek, amikor a hardverzabáló Windows-monstrumok világuralma közepette — lásd például a magyar WinWord diadalmenetét — brilliáns szolgáltatású és csekély helyfoglalású programokról értesülünk. Talán még mindig nem késő alternatívát találni a gigantomániával szemben!

Az amerikai PC Magazine szakírója, J. Dvorak szerint Eric Meyer editora a legjobban megírt szövegszerkesztő. Csupán 47 kbájtnyi helyet foglal, de többablakos, kiválasztható WordStar-, MS Word-, XY-Write- vagy WordPerfect-kompatibilis parancskörnyezetet, makrózást és EGA/VGA üzemmódot kínál!

Teljesen Assembly nyelven írták, ami a kis méreten kívül gyors működést is eredményez. Grafikus karakterek, nemzeti karakterkészletek használhatók (ezek installálásnál is megadhatók, de a VDE a billentyűzetátdefiniáló programokkal is együttműködik). 8 fájl szerkeszthető egyszerre. Kinyomtatás előtt megtekinthető az oldalak hű képe. Makrók használata, automatikus makrórögzítés, elválasztás, élőfejek, margók, sorkiegyenlítés, dupla sorköz, két fájl összehasonlítása, automatikus oldaltördelés stb. jellemzik.

A nyomtatók proporcionális (arányos, változó szélességű) betűit is produkálja. A fontok szélessége megadható, ennek segítségével sorkiegyenlítésnél is figyelembe tudja azokat venni! Támogatja a „Simple", Diablo 630, Epson FX/LQ, HP DeskJet/LaserJet/PacPg 25in1, Okidata uL 82/92/ Pacem, Panasonic KXP1124/KXP 1592, Toshiba P1351/P321SL nyomtatókat. Saját nyomtatónkhoz is hozzáigazíthatjuk a vezérlőkódokkal.

Az 1992-es keltezésű verzió újdonsága a programból is betölthető alternatív konfigurációs állomány (Alt+U), a DESQView-támogatás, és hogy a videomemóriába történő írás letiltható. A képernyőmódok választéka jelentősen bővült. Például 132 oszlopos képernyőmód installálható a hazánkban igen elterjedt Trident SVGA-kártyákon is.

Szövegblokkokat az új verzióban már kijelölhetünk a Shift és a kurzormozgató billentyűk egyidejű lenyomásával is. Az Undelete parancs többlépcsőssé vált, nem csak a legutóbbi törlést hozza vissza. Az állomány-összehasonlító parancs csak a szavakat egyezteti, nem törődik a szövegformázás eltéréseivel. Megjelent a kiterjesztett AT-billentyűzet (F11, F12) támogatása is.

Kereshetünk egész szavakra, a dokumentumokban automatikus végrehajtású makró alkalmazható. Automatikus sorkiegyenlítés, bekezdésformázás, borítéknyomtatás kérhető. Egész szép kis lajstrom, ugyebár!

Vékony Tamás

Küldj egy levelet — megmondom, ki vagy!

Címzett a címke

Mielőtt bárki is nagyképűséggel vádolna a felcím láttán, érdemes körülnéznie otthoni irattárában.
A formai ismérvek (karakterkavalkád vagy igényes, művi nyomtatvány) alapján bárki és nagy biztonsággal eldöntheti, ki küldte az ominózus levelet: ezt az OTP, ezt valamelyik „ágrólszakadt", veszteséges üzletágai elől üvegpalotába menekülő biztosító, emezt az önkormányzat, amazt az adóhivatal — ezt a nagyon rútat meg az Új Alaplap szerkesztősége...

Nem kívánunk abba belemenni, ki mire költi az adó- vagy az előfizetők pénzét, mennyit áldoz ebből a külső megjelenésre, reklámra, mennyit fialtat, és mennyit áldoz a megsarcoltakkal ápolandó kapcsolat minőségi jegyeinek javítására (más lapok más hasábjain bízvást lehet olvasni erről — többnyire az olvasói levelek rovatában), legyen elég annyi, hogy kiküldött leveleinek számítógépi kivitele alapján szinte sohasem az tűnik a legszegényebbnek, aki a leghangosabban szokott siránkozni.

Apropó, siránkozás. Amikor telefonszámlát kapok, vagy a villanyóra-leolvasó tisztel meg látogatásával, az *olvasás* mindig nagy örömmel tölt el: teljes pompájában ott van a számítógép generálta produktumon a magyar nyelv ékes színe-java, nyomát sem lelni az o-ra tolakodó kalapnak (ô), a nagy á helyetti Å-nak, vagy a 852-es valamely gombóc-, illetve farkincaszemetének. Persze, amikor a számokra téved a tekintetem, már csak az egyik szemem nevet...

Amit viszont az OTP-től kaptam, az már a tiszta röhej kategóriája. Először két éve értesített a kerületi számlavezető fiók, hogy devizaszámlámon 4, azaz négy pfennig követelés áll fenn, bármikor rendelkezhetem felőle. A szöveg ékezethelyes, korrekt helyesírással készült, kifogástalan, új festékszalag által nyomattatott. Egy évvel később — merthogy járjon utána a fene mai áron nem egészen 2 forintnak — jött a következő értesítés: ékezetek a helyükön, a 4 pfennig még mindig négy pfennig, s amint az elhomályosuló nyomat alapján éles elmével kikövetkeztettem, a festékszalag is maradt az egy évvel korábbi. Megint egy év, e sorok írójának keresztneve már „makulás": delta jellel ékesített, az utcanév nagy Ö-je gyerekrajzok alagútját formázza, az OTP-fiók pedig kerületiből ker:letivé alakult. Viszont a négy pfennig nem is kamatozott egy huncut garast sem — a háromévnyi postaköltség meg a rezsi és egyebek elvitték az egész nyereséget... Mindenesetre a banki tenderek nyerteseinek egy számlavezető program megalkotásakor arra is illenék gondolniuk, hogy egy ilyen elenyésző nagyságrendet hordozó adat nyilvántartása mennyibe is kerül. Az elhanyagolhatóság mértékét, az azt figyelő rutint elég lenne egyszer megírni.

Következzék egy önfricska. Bevezető mentségül engedtessék meg arra a képtelen géptelenségre hivatkozni, amely az IDG-től eltávoztunkban jellemezte Új Alaplapot alkotó hétköznapjainkat.

Számítógépes lapot szerkeszteni úgy, hogy egy fia gép nem sok, annyi sem áll rendelkezésre a szerkesztőségben, hát... Végül is megjelent a januári szám, nem? Kiküldeni úgy egy jó csomó számlát, hogy a magánhangzók többsége áldozatul esik a nyomtató falánkságának? Ha nem is éppen elegáns megoldás, de legalább volt a környezetünkben egy nyomtató, amely sok-sok betű kinyomtatását lehetővé tette. (Hogy semmi semmivel nem volt kompatibilis, hogy a magánhangzók a legváltozatosabb formában bukkantak fel mindenütt, csak éppen nem a saját helyükön, ráfogjuk a nagy magyar kódkáoszra.) Kézzel címezni nagyon-nagyon sok borítékot? Nem elegáns, de az idő sürgetése miatt már nem volt mit tennünk. Tanulság? Van. Például az, hogy micsoda felelőtlenség hagyni kisöpörni a fogyatékosságai ellenére is egységes CWI-kódkiosztási kultúrát — a nyomtatót rendelkezésre bocsátó alvállalkozónk ugyanis csak azt nyújtotta, amire az ő hardver-szoftver környezete alkalmas volt —, s engedni a 852-es gettó ígéreteinek, hagyni olyan nyomtatóknak piacot nyerni Magyarországon, amelyek sok mindenre alkalmasak — leginkább a kódkáosz elmélyítésére —, csak éppen a magyar nyelven íródott számítógépi produktumok továbbfelhasználhatóságát nem támogatják.

Tessék mondani, mit ér a CWI-kódokkal bezongorázott adatbázis, ha már a shareware-világban sem lelni olyan rutint, amely korrekt módon lenne képes megértetni egy magát FX-kompatibilisnak mondó nyomtatóval, miben különbözik a farkincás C betű a nagy Á-tól? Kutatásaink eredményeként sikerült viszont találnunk egy olyan, Ventura-kódkiosztást letöltő programocskát, amely ezzel az elvarázsolt nyomtatóval is képes szót érteni, s így a címkenyomtatás ezentúl már nem válik éjszakákon át tartó rémálommá, ha... Ha tudniillik sikerül az adatbázis megfelelő mezőinek szövegét CWI-ről Venturára konvertálni. Ez pedig — gondolná bárki — egy magától értetődő Search & Replace funkcióval megoldható.

Aha... Mi is így gondoltuk. Mígnem kiderült, hogy a program ezt a kegyet egyszerre csak egyetlen rekord egyetlen mezőjében hajlandó gyakorolni. Már láttuk magunkat, amint manuálisan átírjuk az egész adatbázist, amikor mentőangyalként segítségünkre sietett egyik külső szerzőnk, és egy apró kis Pascal programocskát megírva szabad szárnyalást engedett dBase-konvertálási vágyainknak. Egyetlen paranccsal minden rekordban és minden mezőben lecseréltük a CWI-kódkiosztással beírt adatokat Ventura-kódszámúra. (Bár ez nekünk nagyobb programozói teljesítménynek tűnt, mint például a Wordstar 2000 több megabájtos konglomerátumának létrehozása, a szerző másként vélekedett, és a dolog bagatell voltára való tekintettel nevének elhallgatását kérte.)

Biztosak vagyunk benne, hogy bármennyire is triviálisnak tűnik a probléma, sokan találkoztak már vele, és fogadják örömmel ezt a kényszer szülte, és a lemezmellékleten közreadott kis programot. Olyanok, akik velünk együtt igyekeznek ellenállni a Windows kínálta és diktálta előnyöknek és kényszereknek, s a gondok ellenére is nagyon jól megvannak a hagyományos, DOS-környezetben.

Varga János

Adatbáziskezelők és a spanyolviasz

Két világ határán

Azzal tisztában vagyok, hogy nem én találtam fel a spanyolviaszt, sőt azzal is, hogy nem én fogok új adatbáziskezelő szabványt kialakítani. Azonban a ma számomra elérhető adatbáziskezelők mindegyikével van valami kis gondom. Hogy mi a gond, és miért éppen a szövegszerkesztőkről szóló összeállításban akarom elrebegni, az a következőkből — úgy vélem — világos lesz.

Tegyük fel, hogy az a felhasználó, aki mindenféle adatait nyilván szeretné tartani, valamint ezek között mindenféle műveleteket szeretne végezni, nem éppen a programozás bajnoka. Sőt, továbbmegyek, nem is ért a számítástechnika lelkivilágához. Az ilyeneknek találták ki a Visual Basic-et, a Lotus 1-2-3-at, meg a hasonló ikszedik generációs fejlesztőeszközöket. De tegyük fel, hogy egyszerűen csak lusta vagyok, vagy az az álmom, hogy használat közben állandóan módosítgassak az adatbáziskezelő rekordszerkezetén, stb. Ez utóbbira jó példa mondjuk az MS Works for Windows (bizonyára a DOS-os változat is jó példa, de azt nem ismerem).

A Worksben egészen jó kis integrált rendszert hoztak össze a fejlesztők, kellően buta minden része, a benne lévő szövegszerkesztő, a spreadsheet (magyarul táblázatkezelő), sőt az adatbáziskezelő is. Azt, hogy kellően buta, nem pejoratív értelemben mondom, hanem csak arról van szó, hogy a legtöbb felhasználónak egy WinWord, vagy egy Excel már túl sok. Az elérhető funkciók nagy részét soha nem fogja használni. Ez kábé olyan, mint annak idején a PL/I programnyelv (vagy később az Ada), amelyeknél kimutatták, hogy egy átlagos programozó az elérhető funkcióknak csak mintegy 30%-át használja, és csak 50-60%-ra tehető azoknak a funkcióknak az aránya, amelyeket egyáltalán ismer a nyelvben. Szóval ennyit a túlbonyolított nyelvekről, és egyéb fejlesztőeszközökről.

Ami viszont a Worksben a pláne, hogy Windows alatt az OLE (Object Linking and Embedding) funkció segítségével teljesen „átjárható" az egész, azaz adatokat mozgathatunk a táblázatkezelő, az adatbáziskezelő, a szövegszerkesztő, és egyéb külső programok között. Persze ez nem a Works érdeme (legalábbis nem a windowsos változaté), hanem magáé a Windowsé.

Na és itt jön a probléma. Az lenne a jó, ha a táblázatkezelő funkcióit, például a kiszámított mezők áldását egy adatbáziskezelő is alkalmazni tudná. Mire is gondolok? Például van a mezei felhasználónak egy otthoni háztartási nyilvántartása a kiadásairól és a bevételeiről. Ezt egyszerűen beüthetné egy táblázatba, egy oszlopban pedig mindig számolná a pénztáregyenleget, és vége. De ha egy kicsit nagyobb igényű az a felhasználó, akkor már lehet, hogy mindenféle kiadásait különböző szempontok szerint is rendezni szeretné, mondjuk mennyi volt eddig a villanyszámlája, vagy mennyit költött a kocsijára, stb. Az olyan funkciók, mint a sorba rendezés (sort), a lekérdezés (query) vagy a beszámolókészítés (report) viszont tipikusan az adatbáziskezelők sajátja. Persze erre könnyen azt lehet mondani, hogy hát akkor miért nem egy adatbáziskezelőt használt az előbbiekben említett „mezei" felhasználó?! Igen ám, csakhogy akkor meg elvész a lényeg, mégpedig az, hogy folyamatosan illik tudni, mennyi is a kassza, mi az egyenleg. Ezt pedig csak táblázatkezelővel lehet kiszámítani.

Éppen ezért az a nagy kérdés, hogy mivel amúgy sincs különbség egy táblázatkezelő cellái, valamint egy adatbáziskezelő mezői között, miért ne lehetne a fent vázolt funkciókat vegyíteni? Végül is pl. a táblázatkezelő C4-es mezőjét lehet úgy tekinteni, mint az adatbáziskezelő 4. rekordjának C nevű mezőjét. Nem véletlen az sem, hogy manapság a fejlett adatbáziskezelők már inkább táblákról beszélnek, úgyhogy biztosan közelít a két funkció egymáshoz, csak még nem találkoztak. Például itt van a Paradox, amely majdnem tudja a fent vázolt igényeket teljesíteni, a baj csak az, hogy kiszámított mezőket (computed fields) csak a „form"-oknál tud kezelni, a táblákban nem. Pedig éppen ez lenne a lényeg!

A Works is majdnem tudja, hogyan lehet táblázatot átvinni az adatbáziskezelőbe (pont a fenti megfeleltetés szerint), csakhogy akkor a számított mezők értékei „fixálódnak", azaz csak az érték megy át az adatbáziskezelőbe, a kiszámítási algoritmus nem.

Visszatérve az adatbáziskezelőkre, biztosan meg lehet oldani a fenti problémát, de itt az a fő, hogy ne kelljen, vagy ne „nagyon" kelljen hozzá programozni (mint például a Paradox Object PAL nyelve, vagy az oly közkedvelt dBase/Clipper, Foxpro stb.).

Persze ezt lehetne tovább is ragozni, pl. a kódkonverzió is egy jó téma lenne. Mi magyarok időnként elég bosszúsak tudunk lenni, hogy az angolszász világ fejlesztői nem igazán tudják a magyar ábécét (pl. az OS/2 is támogatja a magyar nyelvet, csak éppen a hosszú Ú-t nem ismeri).

Sőt ugyanezek a programfejlesztők elfeledkeznek arról, hogy a magyar ábécében az Á betű közvetlenül az A után következik, úgyhogy a sorba rendezés szebben mutatna így, nem pedig az ASCII-szabvány szerint rendezett rekordokkal.

Persze sok adatbáziskezelő tud konverziós táblákat kezelni, de ezek többnyire előre „bele vannak betonozva" az adatbáziskezelőbe, abszolúte nem lehet őket szerkeszteni, miegyéb. (Lásd ismét a Paradoxot, amely nem megy tovább annál, mint hogy a svéd ábécé, vagy valamilyen más 850-es kódtábla szerint lehessen benne rekordokat rendezni.)

Azt hiszem, az előbbi fejtegetésekből már látszik, hogy miről is van szó, és érthetetlen számomra, hogy nincs ilyen integrált rendszer, pedig minden elem megvan a kifejlesztéséhez. Ismert dolgokat használna, legfeljebb egy ilyen integrált rendszer nem lenne villám, már ami a válaszidőket és az adatok elérését illeti. Na bumm, azt mondják, az MS Access sem valami gyors adatbáziskezelő, pedig legalább háromszor annyiba kerül, mint a Works.

A spanyolviaszról pedig igazán nincs fogalmam, hogy mire használják, ha használják egyáltalán valamire.

Sík Zoltán

A honosítás előszobája

„Írhatta volna szebben..."

Ha magyar nyelvű Windowst, Word for Windowst vagy hasonlókat használunk, hamarosan hozzászokunk, hogy a Windows alapjában véve (az esetek 80 százalékában) magyarul üzenget, meglehetősen becsületes magyarsággal. Mégis viszonylag gyakran megtörténik, hogy egyszer csak angolra fordítja a szót, sőt — kapaszkodjunk meg! — olyan is előfordul, hogy egy született és magyarítatlan angol program egyszer csak magyar üzenetet kezd el küldözgetni.

A nyelvek keveredésének oka a Windowsban nagyon egyszerű. A Windows nem katonás elvek (hierarchikus hatáskörök) szerint felépült rendszer, hanem egy eszközhalmaz, ahol az egyik fő elv az, hogy akármelyik munka kellős közepén elő lehessen kapni bármilyen másik eszközt, amelyre éppen szükség van. Ezáltal meglehetősen sok szállal összefonódott rendszer jött létre. Egy felhasználó nemigen tudja kiszámítani, mikor kerül bele a rendszernek olyan ágába, amely még nincs megmagyarítva, vagy a benne lévő program annyira amerikai, hogy a magyar környezetben egészen más kell helyette (pl. valamilyen szótár, betű szerinti rendezés stb.).

Ezen elv előnye, hogy nem kell ugyanazt a programot több változatban megírni, és több példányban beletenni a rendszerbe. A szövegszerkesztőből használhatjuk az adatbázist, és ha az adatbázisból kimutatást készítettünk valamiről, használhatjuk a hozzákapcsolt szöveg megírására a szövegszerkesztőt. És ugyanazt a helyesírás-ellenőrző programot használhatjuk a WinWordből, a Worksből, az Excelből. Persze a dolog csak akkor megy igazán, ha a program eleve beleszületett a Windowsba, és nem később buherálták bele.

Áldás és nyűg: honosítás

Valamivel később, és az előbbi koncepciónak némileg keresztbe téve jött az új eszme: a honosítás. A nálunk nagyobb és gazdagabb országokban, mint Németországban vagy Franciaországban már korábban is komolyabban folyt az alapprogramok, és a népszerű, nagy programok saját nyelvi változatának kifejlesztése. Régóta ismerjük az elveket, hogyan kell ilyen programokat írni, és ezek a megoldások jó illeszkednek a Windows szerkezetébe. De persze azt nem várhatjuk, hogy minden programnak egy csapásra meglegyen az összes nemzeti változata, a magyar nyelv úgy 80 százalékban van készen.

A honosítás nem egyszerű dolog. Egy teljesen közönséges angol nyelvű program sem szokott futni az üzenetfájlok németre fordítása után (mert például a német szavak sokkal hosszabbak, és az üzenetek nem férnek el). Más nyelvekben idétlenül hat, ha a fájlnév az üzenet elejére vagy végére kerül; a névhez biggyesztett végződés függhet a név valamilyen tulajdonságától stb. Majd a 49. nyelv után valószínűleg összeáll az a hatoldalas utasítás, hogy miket kell betartani a fordítás és programírás során ahhoz, hogy az 50. nyelvi változatban ne legyenek különösebb gondok.

A honosítás nem ér végett ott, hogy a hibajelzéseket lecseréljük, változtatandó a számok alakja (tizedesvessző vagy pont), pénzegység, mértékegység (esetleg nem triviális konverzióval), a dátum alakja (pl. hányadik héten, a 0. vagy az 1. héten kezdődik az év; mitől függ ez, van-e rá szabvány).

Külön rémálom az elválasztás, mert némelyik szövegszerkesztő igen makacs, a javítás, beszúrás, újratördelés kapcsán ismét saját alapértelmezés szerint meghatározott angol elválasztását veszi elő.

Ha a felhasználó által beírt szövegek helyességét is ellenőrizni és korrigálni akarjuk, ahhoz nyelvészeti programokra és nagyszótárakra van szükség az adott nyelven.

Menüben már magyar is van

A Windowsban és kapcsolt részeiben gyárilag megoldott a magyar helyesírás-ellenőrzés (a szavak szintjén), az elválasztás, a szinonimaszótár (mindezeket a magyar MorphoLogic cég ké-



— Igen, igen, megkaptuk, már tördelik is a cikkét...

szítette el a Microsoft megrendelésére), de amikor megnézzük az „Eszközök" menüt, találunk egy „Nyelvtan..." című pontot is. Ez a nyelvtani ellenőrző, amely pillanatnyilag hiányzik a magyar változatból, noha a hozzá tartozó üzenetek jelentős része már megvan (magyarul). Angol változatát azonban a magyar Windowsból is lehet működtetni, úgy, hogy a szöveget vagy annak egy részét angol nyelvűnek jelöljük ki. (Ez érdekes módon a formátum mezőben adható meg.) És persze jönnek az üzenetek részben angol, részben magyar nyelven egy ablakon belül is.

Természetesen a magyar nyelvtani ellenőrző is készülőben van, hogy mi lesz benne, azt majd meglátjuk, ha kész lesz. Érdemes azonban megnézni az angol nyelvtani ellenőrzőt, egyrészt hogy lássuk, mit várhatunk el tőle, másrészt hogy lássuk: a magyarban valami más szükséges.

Az angol nyelvtani ellenőrző

A nyelvtani ellenőrzőnek tulajdonképpen két nagy feladatkört kell ellátnia: a nyelvtani hibák felderítését és a stilisztikai hibák kiderítését. A kettő között nemigen érezni világos elhatárolást, hiszen a zsargonszavak használata az elsőbe tartozik, az elavult alakok használata a másodikba. Mindenesetre a második rész kikapcsolható.

Voltaképpen mindkettő egy-egy 20 elemű lista. Az ellenőrzésben 3 fokozat lehetséges: teljes ellenőrzés, üzleti levelek ellenőrzése, gyenge ellenőrzés; valamint felhasználó által választott ellenőrzés. Az első fokozat világos, mindent ellenőriz, a második, harmadik egy-egy előre rögzített, de nem teljes lista, a legutolsó esetben mi jelöljük ki, hogy a kiválasztott ellenőrzési szintből mit akarunk elhagyni, és mit akarunk hozzávenni.

Persze nem lehet mindent kikapcsolni csak úgy ész nélkül (illetve lehet, de nem veszi figyelembe), mert az angol ellenőrzés mindenképpen egy szótárazással indul, megállapítható, hogy mi a szó szófaja, egyes vagy többes számú-e stb. Ha a szó hibás (spelling error), akkor a nyelvtani ellenőrző nem tud róla semmit megállapítani, így a megfelelő nyelvtani egységről sem tud semmi értelmeset mondani. Hasonlóan fontos a másik lépés, hogy megpróbálja a szöveget mondatokra bontani. A felkiáltójel és kérdőjel nem gond, de a pont, akárcsak a magyarban, igen. Szeret például a fájlnév és kiterjesztés közötti ponton megállni, és ilyenkor közli, hogy a ponttól jobbra vagy balra eső rész nem mondat. Az új bekezdés mindenképpen mondatot kezd, függetlenül attól, hogy volt-e a végén valami írásjel. Mivel feltételezi, hogy a szöveg mondatokból áll, nem szereti az olyan címeket, alcímeket, üzeneteket, amelyek nem alkotnak teljes mondatot (például: Out of stack space. Too many errors.), igaz, hogy a Súgó (Help) ilyenkor közli, hogy bizonyos körülmények között helyénvaló is lehet az ilyesmi.

Az igazi finomságok

1. A névszói kifejezés helyessége. A névszói kifejezés végén egy főnév áll, előtte melléknevek, számnevek, kvantorok, névelő stb. állhatnak; ez a pont azt ellenőrzi, hogy a számok után többes szám jön-e, „many", „all" után többes szám, „much", „each" stb. után egyes szám. A magyar névszói kifejezés kissé komplikáltabb lehet, mint az angol, és a főnév gyakran hiányzik. A számok után nem használunk többes számot, sőt kvantorok és kvantor jellegű névmások után sem.

2. Megszámlálható és megszámlálhatatlan. A főnévtől függ, hogy „many"-vel vagy „much"-csal kell a „sok"-at kifejezni, ami egészen angol probléma. A határozatlan névelő használata (a/an) viszont formálisan hasonló a magyarban a határozott névelők (a/az) használatához, a mássalhangzóval illetve magánhangzóval kezdődő szavak előtt.

3. Az igei kifejezés helyessége. Azt az esetet ellenőrzi, amikor az állítmány nem egyetlen ige, hanem egy segédigés szerkezet. Hosszú lenne elmagyarázni, milyen eltérő vélemények vannak a segédigéről magyar nyelvészi körökben, de a „meg kell mondanunk", „ki tudjuk számítani" stb. esetén a „kell" és a „tud" amolyan segédigeféle, ezért nem írjuk az igekötőt vele egybe, hiszen az nem hozzá, hanem a mögötte álló igéhez tartozik. Mindenesetre a mi helyesírási problémáink teljesen különböznek az angolétól.

4. Passzív ige. Igen érdekes, hogy az angol nyelvészek is keményen tiltakoznak a passzív igealak ellen, és azt javasolják, hogy ahol lehet, használjunk aktív formát. Innen Magyarországról ez valahogy mégis kissé másképp néz ki. Mindenesetre normál angol szövegeken meglehetősen gyakran jön be, minden harmadik-negyedik mondatra, de kikapcsolható.

5. Alany—állítmány egyeztetés. Angolban ez azt jelenti, hogy a ragozott igéről (az állítmányról) és az előtte álló főnévről vagy névmásról (az alanyról) kell eldönteni: összeillenek-e; azaz egyes szám harmadik személyű alany és jelen idejű ige mellett van-e az ige végén „s". Nem tudom, hogy az angolok el szokták-e rontani, de én elég gyakran. Magyarban ugyanez sokkal komplikáltabb, a külföldiek mégsem nagyon rontják el. (No persze a tárgyasság vagy a felszólító mód az egészen más!)

6. Here és there egyeztetése. A „Here is...", There is..." kezdetű mondatokkal foglalkozik, ahol az ige előtt álló szó nem az alany.

7. Névmási hiba. Alany helyén alanyesetű, máshol tárgyesetű névmás kell.

8. Gyakran összetévesztett szavak. Hasonló hangzású szavak (ly/lie stb.), amelyek jelentése egész más. Ilyen magyarul is van (pl.: egyelőre — egyenlőre, válság — váltság stb.).

9. Szóhasználati hiba. (Pl.: its/it's stb.)



— Speciális, olvashatatlan fontkészlet, kizárólag receptírásra

10. Ismételt kifejezés. Elsősorban az ... of ... of ... of ... birtokos láncokra ad ilyen hibajelzéseket, az ilyen hosszú birtokos láncok — pl.: a függvény paraméterei értékkészletének maximuma — magyarul sem hangzanak jól, de nálunk számos trükk lehetséges: például „a függvényparaméterek értékkészletének maximuma". Angolban néha nagyobb gond az átalakítás, a Súgó mutat rá példákat.

11. Pongyola kifejezés. Ilyet legfeljebb szóban használunk, írásban nem. Megjegyzés: utál bizonyos aposztrófos alakokat. (Pl.: I'll stb.)

12. Nem hivatalos alak. Magyarban ilyesminek felel meg: máma — ma helyett, engemet — engem helyett stb.

13. Kétszeres tagadás. Az angol mondatban szigorúan csak vagy egy bővítményt lehet tagadni, vagy az igét. Magyarban a tagadó névmások mellett kötelező az igét is tagadni, más tagadás mellett tilos. Például: „Senkit nem láttam", de „Nem Pétert láttam". A formális logikán nevelődött emberek hajlamosak többszörös tagadást is használni — pl. Nem Pétert nem láttam, hanem... —, ez sokak számára érthetetlen, zavaros beszéd.

14. Zsargonszó. Ha a teljes ellenőrzés van beállítva, a program feltételezi, hogy irodalmi szövegről van szó, ilyenkor sűrűn osztogatja ezt a hibajelzést az interfész, a paraméter és hasonló szavakért, azzal a megjegyzéssel, csúnya technikai zsargon, időnként ajánl is helyette valami egészen nem odaillőt.

15. Nem hivatalos módosítószó. Akkor osztogatja, ha módhatározóként melléknevet és nem határozószót akarunk használni. (Pl.: They automatic adjust their speed. „Automatic" helyett „automatically". Lásd magyarban: halál szép, halálosan helyett.)

16. Formátumhiba. (Dátum, szám, stb. ellenőrzése.)

17. Mondathiba. A szavakból nem tudott mondatot összerakni. Megjegyzés: másfél oldalnyi szövegben egyetlen olyan mondat akadt, amely nem alany-állítmánnyal kezdődött, hanem egy előrevetett határozóval, és ezt az egyet rögtön kiszúrta.

18. Írásjelhiba. Ez elég nehéz ügy, de úgy tűnik, jól csinálja, vesszőre, pontosvesszőre elég értelmes magyarázatokat és példákat ad. Nem tetszik viszont, hogy az idézőjelbe, zárójelbe tett részben is úgy kezeli az írásjeleket, mint a főmondatban. A pontot éppúgy mondatvégnek veszi, mintha az idézőjel ott sem lenne. A szavak közé tett „ / " jel (pl.: input/output) helyett javasolja a „vagy", „és", „illetve" szavakat.

19. Szóhiba. Nem ismeri a szót. Az elválasztást és nagybetűt is itt vizsgálja.

20. Rendszerüzenet. Túl hosszú szó, túl hosszú mondat. Valami meghaladja a lefoglalt pufferméretet. A sormintákat nem szereti.

Stilisztikai hibák

1. Szószaporítás. Több szó használata egy helyett, pl.: „it met your need" — helyett „satisfy". Magyarban is van ilyen: „átvételre kerül" — „átveszik" helyett, „befejezést nyer" — „befejeződik" helyett stb.

2. Redundáns kifejezés. Egy része felesleges: „very latest". Magyarban is van ilyen: „legminimálisabb" — holott: minimális = legkisebb.

3. Rossz prepozíció. Fölösleges vagy nem odaillő elöljárószó. Az igétől függ, mikor milyen elöljárót kell használni. Ez angolban sem egyszerű, bár pl. a Hornby-szótárban egy (vagy több) kód mutatja, hogy az igének milyen vonzatai lehetnek.

4. Pongyola kifejezés. Amely szóban használható, de írásban nem.

5. Nyelvi klisék/sablonok. Magyarban is van ilyen: őszintén szólva, színpompás felvonulás stb.

6. Bizonytalan módosítószók. A melléknevek és az igék előtt álló módosítószók köre meglehetősen szűk, pl.: elég jó, alig látom, nemigen jön, állítólag jön, bizonyára szép stb. Véleményem szerint irodalmi, különösen lírai szövegekben ennél sokkal több határozószó és hasonló állhat az ige vagy a melléknév előtt. A nyelvtani ellenőrző mindenesetre valamilyen egzaktsági elvtől vezérelve nem szereti pl. a „great difference" és hasonló alakokat.

7. Bizonytalan kvantor. Magyarban is van ilyen: majdnem mind, meglehetősen sokszor, nagyon ritkán stb., amikor eltöprenghetünk azon, hogy valójában mennyi is az annyi. Használatukat különösen technikai leírásokban kell kerülni.

8. Hibás kifejezés. Olyan kifejezés, amelyet a Standard English nem használ. Nem tudom, mi ez, a „gonna" alakot mindenesetre eszi.

9. Egybe- és különírás. Ha van valami probléma a magyarban, hát ez az: vöröshagyma — vörös káposzta, Mátraalja — Pilis alji stb. Ehhez képest az angolban felmerülő egybe- vagy különírási gondok eltörpülnek.

10. Többszörös tagadás. Egynél több negatív mondatrész a mondatban, pl.: „nem megoldhatatlan". Ez az angolban súlyos stílushiba.

11. Helytelen kifejezés. Rosszul leírt kifejezések.

12. Rosszul használt szó. Olyan szavak, amelyeket gyakran írnak rosszul.

13. Zsargonkifejezés. Nem a nagyközönségnek való szavak és kifejezések. Sajnos nem lehet megjelölni, hogy szépirodalomról van-e szó, vagy technikáról, gazdaságról stb. Szerencsére ki lehet kapcsolni.

14. Fellengzős kifejezés. Komplikált szavak/kifejezések helyett egyszerűbbet ajánl, pl.: „all of the" helyett „all". Úgy tűnik, ízlés dolga, hogy mi tartozik ide, mi a redundáns alakhoz, és mi a szószaporításhoz.



— Maga magyar kódexmásoló és csak svéd karakterekkel tud írni?

15. Elavult kifejezés.

16. Rosszul leírt idegen kifejezés. Közismert latin, francia stb. kifejezések ellenőrzése pl.: ab ovo, c'est la vie stb.

17. Túlhasznált kifejezés. Túl gyakran használt ige, főnév, melléknév, határozószó. Ez megint inkább a szépirodalomban hiba. Technikai szövegekben igen fontos, hogy valamit mindig ugyanannak nevezzünk.

18. Kihagyható kifejezés. Olyan jelző, határozószó stb., melynek elhagyása nem változtat a mondat értelmén.

19. Az idézőjelbe tett szöveget is ellenőrizze. Sajnos nem igazán lehet kikapcsolni. A mondathatár keresésénél nem nézi, hogyan intézkedtünk.

Paraméterek

A fentieken kívül három paramétert lehet beállítani a Catch címszó alatt:

— *Hány bővítménye lehet a főnévi igenévnek a mondatban.* Elvileg akárhány lehet, de az esetek nagy részében csak egy van. Pl.: Elmentem a városba könyvet venni. (A „városba" az „elmentem" szónak, viszont a „könyvet" a „venni" szónak a bővítménye, és mindkettőnek lehetne még további bővítménye is.) Nem tudom, miért köti meg a számot.

— *Hány főnév állhat egymás után.* Egy névszói kifejezésben normálisan egyetlen főnév áll, no persze csak akkor, ha a tulajdonnevet, anyagnevet stb. nem tekintjük főnévnek. Pl.: Kovács tanár úr. Ha tehát három főnév áll egymás után, úgy, hogy ragokkal (angolban prepozíciókkal) vagy írásjelekkel sincsenek elválasztva, az már gyanús.

— *Hány elöljárós kifejezés állhat egymás után.* Az angolok úgy tartják, hogy a túl sok elöljárós kifejezést tartalmazó mondat nehezen érthető.

WinWord



A fentieket beállíthatjuk az ellenőrzés előtt, vagy ha valamilyen jogosulatlannak érzett hibajelzés túl gyakran fellép, egy hibajelzés után is.

Mindezek után indulhat az ellenőrzés. Amikor a program hibát talál, hibajelzést ad. A hibajelző ablaknak három fő része van:

- A hibás szövegrész. (Szó, kifejezés vagy mondat.)
- A hibajelzés és módosítási javaslat. Ha a szöveg ugyanolyan — esetemben fehér alapon fekete —, mint a hibás szöveg, akkor az egy ajánlat: olyan kifejezés, amelyre javasolja kicserélni a hibás alakot. Ha nem ilyen — nálam kék alapon fehér —, akkor csak hibajelzés.
- Részletesebb magyarázatot is kaphatunk egy új ablakban, példákkal fűszerezve. Ezt az ablakot sajnos be kell csukni, hogy intézkedhessünk.

Intézkedések

— Az egyik ilyen, hogy magyarázatot kérünk.

— A másik, hogy negligáljuk a hibát, ilyenkor a mondatban lévő következő hiba ugrik be. (Ez persze, akárcsak a programozási nyelveknél, lehet az előző következménye.)

— A harmadik utasítás, hogy menjünk a következő hibás mondatra.

— A negyedik, hogy javítunk.

— Az ötödik, hogy módosítjuk, mit kell ellenőrizni.

Ha sikerült a szöveg végére érni, egy takaros kis táblázatot kapunk a szöveg olvashatóságáról. Adatok vannak benne a szavak, mondatok, bekezdések számáról és átlagos hosszáról; a passzív mondatok arányáról és hasonlókról. És három mutatót is kapunk, amely a szöveg olvashatóságát numerikusan fejezi ki. Az angoloknak vagy öt ilyen mutatójuk is van, a WinWord különböző verziói más és más mutatókat számolnak ki.

Nem tudok róla, hogy magyar nyelvvédő valaha is ilyen numerikus mutatóval próbálta volna kifejezni a magyar szöveg érthetőségét, olvashatóságát.

Következtetések

Először is le kell szögeznünk, hogy egy szöveg nyelvtani helyességét, szépségét vagy zavarosságát csak akkor tudjuk megítélni, ha megértettük. A nyelvtani ellenőrző nem érti meg a szöveget, csak bizonyos formai dolgokat tud ellenőrizni. Minden harmadik-negyedik esetben talál a probléma közelébe, és legfeljebb minden tizedik

Lassú szövegszerkesztő



esetben ad teljesen odaillő javaslatot, még a legjobb beállítás mellett is. Én a programot három eredeti amerikai program dokumentációs állományán próbálgattam (a nyelvet természetesen US angolra állítva). A szenvedő szerkezet kikapcsolása után egy oldalon átlagosan öt hibajelzés jött. Azok kétharmada egyszerűen tévedés volt, de szinte minden lapra jutott egy „mondhatta volna szebben, kis lovag" jellegű intelem. Voltak valódi hibák is, például mondat alany és állítmány nélkül. (Az „It is..." hiányzott.)

A fentiekből az látható, hogy a nyelvtani ellenőrző tulajdonképpen egy hagyományos nyelvvédő nyelvtanárt imitál.

Elképzelhető lenne egy olyan nyelvhelyesség-ellenőrző program, amely a szöveget az angol nyelv valamilyen gépi matematikai modelljével veti egybe, és ami beleillik, elfogadja, ami nem, arra hibát jelez. Itt nem erről van szó.

— Egyrészt vannak sémák, amelyeket felismer, és azokból egyeseket szeret (alany, állítmány, indirekt tárgy, direkt tárgy, elöljárós tárgyak vagy „there is..."); másokat felismer, de nem szeret (szenvedő mondat); míg megint másokra fel sem készül, olyan ritkának tartja őket (előrevetett határozó).

— Másrészt vannak elvei: a túl sok határozót tartalmazó mondat nehezen érthető, a szóismétlés nem szép, a szöveg legyen pontos és egyértelmű.

— Harmadszor készletei vannak a szokásos közhelyekből, képzavarokból, stílustalanságokból, elírásokból stb., ezeket keresi a szövegben. Ezek több csoportba vannak osztva, és egyenként kikapcsolhatók. Sajnos a beosztás kritériuma sehol nincs leírva, mitől szószaporítás valami, és miért nem redundáns vagy fellengzős kifejezés, pedig mennyivel könnyebb lenne használni.

Lenne egy-két ötletem, hogyan lehetne ugyanezt egy kicsit ügyesebben megcsinálni a magyarra. Persze kérdés, hogy az interfész engedné-e.

Farkas Ernő

Szöveg a Unix ege alatt

Hogyan szerkesztünk „odaát”?

Ha az ember Unix alatt ír egy levelet, jelentést, vagy akár csak egy aprócska programot, gyakran kerül szembe komoly problémákkal: a Unix szabványos szövegszerkesztői ugyanis nem éppen felhasználóbarátságukról híresek.

A sorban az első az ed, amely kicsi, gyors — ámde van egy rendkívül kellemetlen tulajdonsága: ún. soreditor (azaz mindig csak egy adott sor szerkesztésére van lehetőség — ráadásul nincs semmiféle soron belüli pozicionálást, vagy akár egy sor közepén lévő karakter törlését egyszerűen lehetővé tevő parancs).

Persze vannak előnyei (ha nem is sok), de a mai PC-s világból érkezőket elég nehéz meggyőzni ezekről az előnyökről. A következő editor az ex, amely úgyszintén soreditor, bár vannak hasznos fejlesztések benne. A szabványos szerkesztők sorát a vi zárja. Erről egyszerűen csak annyit kell tudni, hogy szakmabeliek szerint „pilótavizsgás” program. (Vagyis — ha valaki nem tudná — hogy kb. akkora erőfeszítéssel lehet megtanulni a használatát, mint mondjuk a helikoptervezetést, igaz, ez utóbbit tudtommal senki nem próbálta azok közül, akik ezt hangoztatják.)

És az emacs?

A vi azonban elterjedt, és minden hibája ellenére (bár nem sok van neki) használják. Nagy unixosok rögtön közbevághatnak: és az emacs? Ez egy public domain szövegszerkesztő — amely viszont nincs meg minden rendszerben. És ezzel el is árultuk legfontosabb hibáját.

Az első komoly problémát az szokta okozni, hogy noha a vi teljes képernyős szerkesztő, a kurzormozgató billentyűk vagy egyáltalán nem működnek, vagy igen rejtélyesen. Sajnos rögtön az elején el kell árulnunk, hogy (főleg a régebbi Unix-változatokban) a vi semmilyen módon nincs felkészítve a kurzormozgató billentyűk kezelésére, s ezekben a rendszerekben kevés esélyünk van arra, hogy valóban jól lehessen kurzormozgató billentyűkkel kezelni a vi-t. Azt, hogy legalább elméletileg ismeri-e a nyílbillentyűket, például úgy tudjuk megállapítani, hogy a vi elindítása után kiadjuk a :map parancsot. Ha az így kapott listában megtaláljuk az up, down, left, right makrókat, akkor a vi (szinte biztosan) tudja használni őket. (Ebből a listából az is látszik, hogy a vi szerint milyen karaktereket küldenek az egyes nyílgombok.)

A dolog megértéséhez kicsit vissza kell menni az időben, és néhány technikainak látszó dolgot el kell mesélni. Amikor a vi-t írták, (úgy ’75 környékén), a számítógépek általában terminálokkal működtek, amelyeken nem nagyon voltak kurzormozgató billentyűk. Ezért nem készült fel a vi (akkor) arra, hogy használja azokat. (Azon a terminálon, ahol a vi fejlesztője dolgozott, a ’hjkl’ billentyűkre volt felgravírozva a négy kis nyilacska — ezért használja ezeket a billentyűket a vi a kurzor mozgatására.) Később egyre újabb és újabb terminálok jelentek meg a piacon, amelyek természetesen anynyira voltak egységesek, mint mondjuk manapság az SVGA-kártyák.

Nincs olyan gomb!

Az adott probléma szempontjából nézve a dolgokat: a terminálokon megjelenő kurzormozgató billentyűk olyan kódot (illetve kódsorozatot) küldtek, amilyen a terminál gyártójának tetszett. Ez azonban még egy adott gyártó esetében sem volt egységes, elég csak az egyik kvázi-szabványt kidolgozó DEC termináljait megnézni. A vt-52 típusjelű DEC-terminál például a négy kurzormozgató billentyű megnyomásakor az <ESC>A (föl), <ESC>B (le), <ESC>C (jobbra) és az <ESC>D (balra) kétkarakteres szekvenciákat küldte. A vt-100 jelű terminálján pedig ezt továbbfejlesztette, és az <ESC> és a betűk közé beszúrt még egy karaktert: [-t (szögletes zárójelet) normál, és O-t (nagy O betűt) az ún. application keypad módban (ez a kettő olyasmi, mint a normál PC-billentyűzet numerikus blokkja NumLock ON és NumLock OFF módban). Ha ehhez hozzávesszük azt is, hogy a vi az <ESC> karaktert igen komolyan használja (és a vt-52-n és vt-100-on nincs is olyan gomb, amely ezt küldené!), akkor látható, hogy a probléma megoldása nem túl egyszerű.

Jelenleg az előforduló terminálok és terminálemulátor-programok 99%-a tudja a vt-52, vt-100, vt-220 stb. sorozat valamelyikét (vagy többet közülük). Azaz pl. a kurzormozgató billentyűk valószínűleg e sorozatok valamelyikét küldik.

Ha a vi olyan régi, hogy még nincs beleépítve a kurzormozgató billentyűk kezelése, meg lehet próbálkozni a következőkkel: létre kell hozni 4 ún. billentyűmakrót (lehet, hogy sokak számára meglepő, de a vi makrókat is tud kezelni!). A megoldás a következő: be kell gépelni a következő parancsokat:

```
:map ^V<föl-nyíl> k
:map ^V<le-nyíl> j
:map ^V<bal-nyíl> h
:map ^V<jobb-nyíl> l
```

A ^V azt jelenti, hogy begépelünk egy kontroll-v karaktert (ilyenkor egy ^ karakter jelenik meg a képernyőn, és a kurzor azon áll). A <..> pedig a megfelelő kurzormozgató billentyű lenyomását jelképezi.

Ezzel elértük azt, hogy pl. a <le-nyíl> által elküldött karaktereket a vi a j karakterrel helyettesítse (parancsmódban ez a „le” parancs). (Noha a vi-ben a beszúró módban a kurzor mozgatása akkor sem javasolt, ha a vi képes azt kezelni, meg lehet próbálkozni olyan makrókat írni, melyek ugyanezen a módon „átdefiniálják” a billentyűzetet. De némi átgondolást igényel a dolog!)

Termcap és terminfo

Térjünk át arra az esetre, ha a vi tudná kezelni a nyílgombokat (a lehetőség tehát adott), de valamiért mégsem működik. Ekkor valószínűleg a rendszer valamilyen adatbázisa nem megfelelően írja le az általunk használt terminált.

A millió különböző terminál kezelését valahogy meg kellett oldani. Ezt a Unix a termcap (illetve az újabb verziókban a terminfo) nevű adatbázisokkal oldja meg. A vi pedig ezeket az adatbázisokat használja. Ráadásul az sem egységes, hogy melyiket: bizonyos Unix-verziókban a termcapet, bizonyosakban (inkább az újabbakban) a terminfót. Ebből viszont az következik, hogy egy új (vagy akár egy kompatibilisnak kikiáltott) terminál rendszerbe állításakor ezeket is rendesen be kell állítani. A beállítás apró pontatlanságai hatalmas(nak látszó) gondokat okoznak a használatban.

Ilyen pl. az, hogy nem megy a kurzormozgató billentyű. Elsőként nézzük meg, hogy az éppen aktuálisan használt terminálon mit küldenek a kurzormozgató billentyűk. Ezt nagyon sokféle módon meg lehet tenni, most itt csak egy módszert ismertetünk — természetesen a vi segítségével. Lépjünk be a vi-be, és váltsunk át beszúró üzemmódra. Gépeljünk be egy ^V-t (kontroll-v), utána pedig (mondjuk) a kurzor-fel gombot. Az eredmény valami ilyesmi lesz: ^[[A. (A ^[az <ESC> karakter vi-beli megjelenési formája.) A fenti eszmefuttatás alapján elképzelhető még a ^[A, illetve a ^[OA sorozat is. (Persze bármi más is, ebben az esetben az itt leírtak némi megszorítással lesznek csak igazak a rendszerünkben.) Ugyanilyen módon a többi kurzormozgatót is nézzük meg (^V+kurzormozgató), írjuk föl ezeket (illetve, ha bízunk a fejünkben, erre nincs szükség), és lépjünk ki a vi-ből.

Következő lépésként meg kell tudni, hogy az adott rendszerben a vi melyik terminálleíró adatbázist használja (ha van mind a kettő). Legegyszerűbb módszer a 'man vi' parancs, amely a kézikönyvlap végén szinte biztosan feltünteti ezt, a Files részben.

Ha ezen is túlestünk, akkor már csak az adatbázis ellenőrzése van hátra (és esetleg a javítása — hmm!). Ha a vi a termcap adatbázist használja, akkor a /etc/termcap nevű fájl kell nekünk (legalábbis szinte biztosan). Ennek formátuma elég ehetetlen, de legalább biztosan forrásban van meg. Ebben a fájlban meg kell keresni az általunk használt terminálhoz tartozó információkat. Azt, hogy milyen terminálon dolgozunk, a TERM nevű változóból veszi a rendszer. Utána pedig a ku (KeyUp — föl), kd (KeyDown — le), kl (KeyLeft — balra) és kr (KeyRight — jobbra) bejegyzéseket kell az általunk megjegyzettekkel összehasonlítani — és különbség esetén javítani.

Rendszerközelben — óvatosan

Egy jó tanács: mivel általában nem tesz jót, ha a rendszer adatait kényünk-kedvünk szerint átírogatjuk, kihasználhatjuk azt is, hogy ha egy program a termcap adatbázist használja, akkor először egy TERMCAP nevű környezeti változót vizsgál, és ha létezik, akkor annak tartalmát használja a /etc/termcap helyett — ha a változó tartalma / karakterrel kezdődik, akkor az így meghatározott fájlt olvassa, ha pedig nem /-rel kezdődik, akkor termcap-beli bejegyzésnek tekinti. Ilyen módon tesztelhetjük az általunk írt terminálleírókat anélkül, hogy a rendszer adatait tönkretennénk.

Ha a rendszerünkben a vi a terminfo adatbázist használja, nehezebb a dolgunk. Ez ugyanis egy bináris adatbázis, és bizonyos rendszerekben a forrás nincs is benne. Az adatbázis (általában) a /usr/lib/terminfo könyvtárban és alkönyvtáraiban található. Ha mázlink van, itt a forrás (pl. terminfo.src néven). Ha nincs meg a forrás, kicsit nehezebb a dolog, ugyanis csinálnunk kell egyet — legalább az adott terminálhoz. Ehhez két segítségünk lehet: egyrészt a captoinfo nevű program, amely — mint neve is mutatja — a termcap adatbázisból terminfo adatbázist 'fordít'. (Miután a termcap adatbázis kevesebb képesség leírására használható, ráadásul a programot is csak emberek írták, tehát lehetnek benne hibák, ha van rá mód, a második módszert alkalmazzuk.)

A másik, az infocmp nevű program a bináris terminfo adatbázist visszafordítja terminfo forrásba. Ha valamilyen módszerrel hozzájutottunk egy terminfo forráshoz, ezt azután már lehet javítani, majd a rendszerben levő tic (terminfo compiler) programmal le lehet fordítani.

A kurzorhoz visszatérve, terminfo esetén a keresendő négy bejegyzés a kcuu1 (Key Cursor Up 1 — kurzor föl egyet), kcud1 (Key Cursor Down 1 — kurzor le egyet), kcub1 (Key Cursor Back 1 — kurzor visszafelé egyet) és a kcuf1 (Key Cursor Forward 1 — kurzor előre egyet). Az ezekben lévő karaktersorozatoknak meg kell egyezniük azzal, amit a terminálunk küld.

Megfelelő javítás után ezt is a rendszerbe illeszthetjük, de a termcapnél mondottak itt is érvényesek: a rendszer adatait ne írogassuk felül, ha nem muszáj. Ezt terminfonál úgy tudjuk kikerülni, hogy létrehozunk egy TERMINFO nevű környezeti változót, amely egy katalógusfára mutat. Ez a katalógusfa szerkezetében ugyanolyan, mint a terminfo adatbázis eredeti (/usr/lib/terminfo) fája. Azaz mathilda-2 típusú terminálunk (lásd TERM változó!) esetén a leírás a TERMINFO változó által mutatott könyvtár m/mathilda-2 nevű állományában található. Vigyázzunk, mert ha nincs TERMINFO változó, a tic az eredeti terminfo adatbázisba ír, úgyhogy ennek beállításával kell kezdeni.

Furcsa nyílgombok

Érdemes megjegyezni, hogy a vi a mozgatóbillentyűket makróként kezeli. Ebből adódhat még egy probléma: mindezeket megcsináltuk, de a nyílgombok furcsán viselkednek. Különösen soros terminálokon (de főleg terminálemulátoron) gyakori az a probléma, hogy néha működnek ezek a gombok is, néha meg nem, sőt mindenféle szemetet írnak a szövegbe. Ezt az okozza, hogy a vi, ha többkarakteres makrókról van szó, akkor az egyes karakterek beérkezése közben egy bizonyos ideig vár.

Ha az idő lejár, mielőtt megérkezne a következő karakter, akkor ezt nem makróként próbálja meg értelmezni, hanem a beérkező karaktereket magukat. Pl. vt-100, kurzor fel: <Esc>[A. Ha ez a bizonyos time-out az A beérkezése előtt lejár, akkor külön parancs az <Esc> (kilépés parancs üzemmódba, ha beszúró módban vagyunk, illetve hiba, és ennek eredményeként sípolás vagy képernyő-villogtatás, ha parancs üzemmódban voltunk már előzőleg is). Ezek után parancs a [, valamint a valamikor később beérkező A is. No de a [magában nem parancs (csak kettőzve), így újabb hibajelzés az eredmény. Az utolsó karakter pedig az A, amelynek jelentése: hozzáadás az aktuális sor végéhez. (Ennek hatására pedig már beszúró üzemmódba is kerültünk.)

Ezt pedig úgy tudjuk kikerülni, hogy kikapcsoljuk a time-out-ot a :set notimeout paranccsal. Vigyázat! Ennek az lesz a mellékhatása, hogy ha magát az <Esc> gombot nyomjuk meg, akkor is a végtelenségig vár a vi (hiszen a nyílgombok is <Esc> kezdetű sorozatot küldenek).

No hát ennyit a vi szépségeiről. Ha valaki a DOS-on nevelkedettek közül kedvet érez egy kis ismerkedésre, a lemezmellékleten található egy elvis nevű program, a vi public domain verziója. Viszont a cikkben leírtak többségét csak Unix rendszer alatt lehet kipróbálni — mivel az MS-DOS-ban nincs sem termcap, sem terminfo.

Zahemszky Gábor

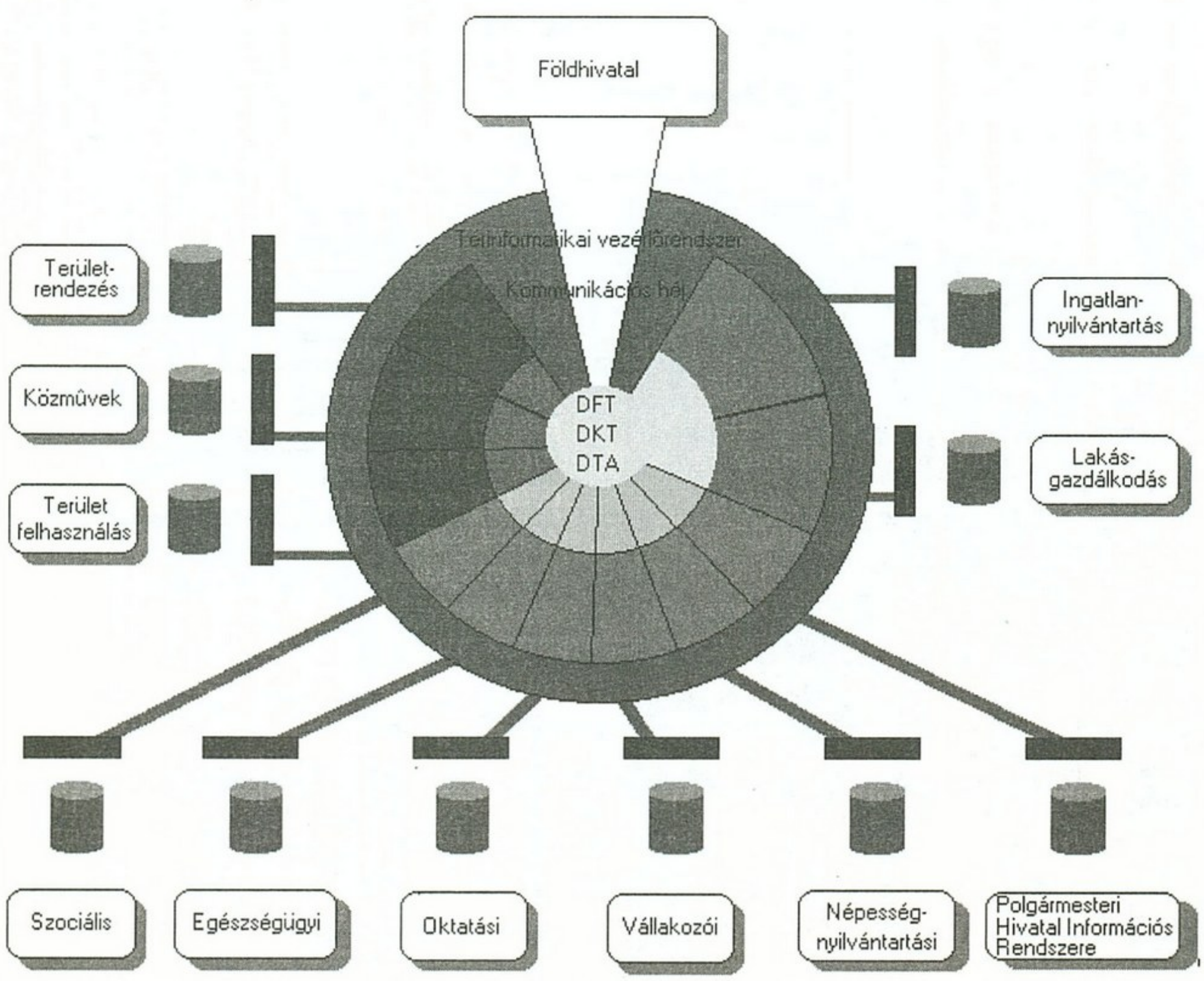
ÖTR = Önkormányzati Térinformatikai Rendszer

Feltérképezett élettér

Bár a térinformatika nem tartozik szorosan a CAD/CAM világához, a távoli rokonság okán érdemes elidőzni egy kicsit ennél a témánál.
Annál is inkább, mert az OMFB térinformatikai nemzeti projektje keretében több város is pályázatot hirdetett önkormányzati térinformatikai rendszerének kidolgozására.
Ezúttal nem egy, a témát reprezentáló szoftvert vizsgálunk meg közelebbről, hanem az elvek szintjén maradunk. „Szoftverfüggetlenül" vesszük sorra azokat a legfontosabb szempontokat, amelyeknek az ÖTR-eknek meg kell felelniük.
Ezek alapján már csak meg kell találni azt a szoftvert, amely eleget tesz a rendszermodellnek.

A számítógépes rendszer a Digitális földmérési alaptérképre (DFAT-re) és az önkormányzati szakadatbázisok adataira épül. A térképi és szöveges adatok együttes és komplex kezelése lehetőséget teremt a helyhez kötött objektumok nyilvántartására, kezelésére és ehhez kapcsolódó magas szintű információk szolgáltatására. Célszerű a programrendszert úgy kialakítani, hogy azt integrálni lehessen egyrészt a polgármesteri hivatalok információs rendszeréhez, másrészt az önkormányzatok tervező, elemző, döntés-előkészítő és napi irányításának munkáját támogató rendszerekhez.

Az ÖTR és környezetének globális adatmodellje a térinformatikai adatbázis szempontjából négy szintre tagolódik.



A globális adatmodell

Az alapadatbázist a legbelső szinten a közterület-tartalommal bedolgozott DFAT és a hozzá kapcsolódó ingatlankataszter képezi. Erre az adatbázisra szinte valamennyi szakági alrendszernek — átnézeti vagy részletes formában — szüksége van, szabályrendszeréhez igazodnia kell. (Az alapadatbázis karbantartását értelemszerűen a Földhivatal látja el.)

Az adatbázis következő (második) szintjét az egyes alrendszerek által az alapadatbázisba fedvényként véglegesen vagy időlegesen bevitt térképi tartalmak jelentik. Ettől a szinttől a térinformatikai adatbázis három funkcionális részre tagolható:

— Geometriailag pontos, véglegesen bent maradó fedvények (például a közmű tartalom).

— Geometriailag pontos átmeneti fedvények, amelyek döntéstől függően később átkerülhetnek az első szintre.

— Pontosságra nem törekvő tematikus térképek.

Az adatbázis második szintjének adatköre az önkormányzati alrendszerek esetében teljes, míg a külső rendszerek esetében alku és előírások függvénye. A szint karbantartásáért az adattulajdonosok felelősek.

Az adatbázis harmadik szintjén — a második szint térképeinek elválaszthatatlan részét képező — szöveges adatállományok szerepelnek, amelyek az egyes szakági rendszerek specifikus adatait tartalmazzák, azokkal közvetlenül geometriai relációba hozhatók (népességnyilvántartás alapadatai). A szint adatkörének pontos kialakítása a közös adatbázis megfogalmazóinak feladata, karbantartásáért viszont a vonatkozó szakági alrendszer felelős.

A szakági adatbázisok (negyedik szint) már nem tartoznak az ÖTR hatókörébe. A szakági rendszereket működtetők döntik el, hogy ebből a szintből mi és milyen hozzáférési jogosultsággal kerül majd a harmadik szintre.

A funkcionális modell

Az ÖTR funkcionális modellje ugyancsak szintekre tagolható: térinfor-

matikai vezérlőrendszer és kommunikációs héj (ez a funkcionális mag); az alaptérkép (DFAT) karbantartása és kezelése; interfész felületek.

Az ÖTR funkcionális magja biztosítja és felügyeli a rendszer logikai és fizikai integritását, tartja a kapcsolatot az egyes szakági alrendszerek specifikus interfészfelületeivel. Vezérli az alrendszerek közötti adatkommunikációt, ütemezi, ellenőrzi és naplózza a hozzáférést a központi és az alrendszerspecifikus térinformatikai adatbázisok között.

A DFAT-hez kapcsolódik és azzal mindig konzisztens állapotban kell lennie a Digitális térinformatikai alapadatbázis ingatlankataszteri részének. Erre a térképre külön térképként kerül a Digitális közműtérkép (DKT) tartalma. Fontos megjegyeznünk, hogy a két térkép ugyan logikailag egyetlen egységet alkot, aktualitásukban azonban különböznek egymástól.

Míg a DFAT a jogi állapotot tükrözi (közhiteles), addig a DKT tartalmilag bővített, a természetbeni állapotot jobban közelítő, de nem közhiteles nézete ugyanannak a területnek. A kettő közötti szükségszerű eltérés a „keltezésükből" fakad.

A digitális térképek az állapottól függően műszaki átdolgozásra szorulhatnak. Az egyes térképi objektumoknak külön rétegekre kell kerülniük, amelyeket azután az alkalmazási témáknak megfelelő témacsoportokba kell fűzni. A térkép intelligenciaszintje megnövelhető az ingatlankataszterrel való egybedolgozással, valamint a geometriai objektumok átalakításával.

A térkép karbantartása változásjelentéseken keresztül valósul meg. A teljes térképből a rendszer automatikusan képez tematikus térképeket, amelyek az egyes szakági rendszerek feldolgozásainak alapjául szolgálnak.

A szakági rendszerekkel a vezérlőrendszer ún. szűrőkön keresztül kommunikál. Ezek a szűrők, bár alrendszerspecifikusak, mégis egységes elvek és mechanizmusok alapján működnek. Egyrészt lehetővé teszik, hogy a szakági alrendszer használója munkájának megzavarása nélkül hozzáférhessen a térinformatikai adatbázis adataihoz, annak azt a szeletét lássa, amelyre feladatának végrehajtásához szüksége van. Másrészt szűri a szakági adatokat a térinformatikai adatbázisba való bekerüléskor.

Adott esetben ez a szűrő egy „ablakká" is redukálódhat, ha nincs tényleges adatforgalom az alrendszer és a térinformatikai adatbázis között.

A funkcionális csoportok

A specifikus szűrőkön kívül általános szűrők is működnek, amelyek a szakági rendszerek számára az azonos térképi tartalmú, színezett térképeket állítják elő.

Az adatmodellek kialakítása után nézzük meg, hogy milyen funkcionális csoportok felelnek meg ezeknek a modelleknek!

A geometriailag pontos állandó fedvényekkel dolgozó alrendszerek feladata az alapadatbázis karbantartása. Ilyen például a területfelhasználási, területhasznosítási alrendszer — ez valamennyi alaptérképi típussal és adatbázissal dolgozik —, mely elsősorban a közterületi tartalomra koncentrál. A közműalrendszerek interfészeinek meghatározásakor a minimális követelményt a nyomvonalszakaszokra és objektumokra (gépházak, transzformátorok, szerelvények stb.) bontott integrált adatszerkezetben kell megszabni. Az adatcsere-formátum minimumkövetelménye: DXF és dBase szövegfájlok.

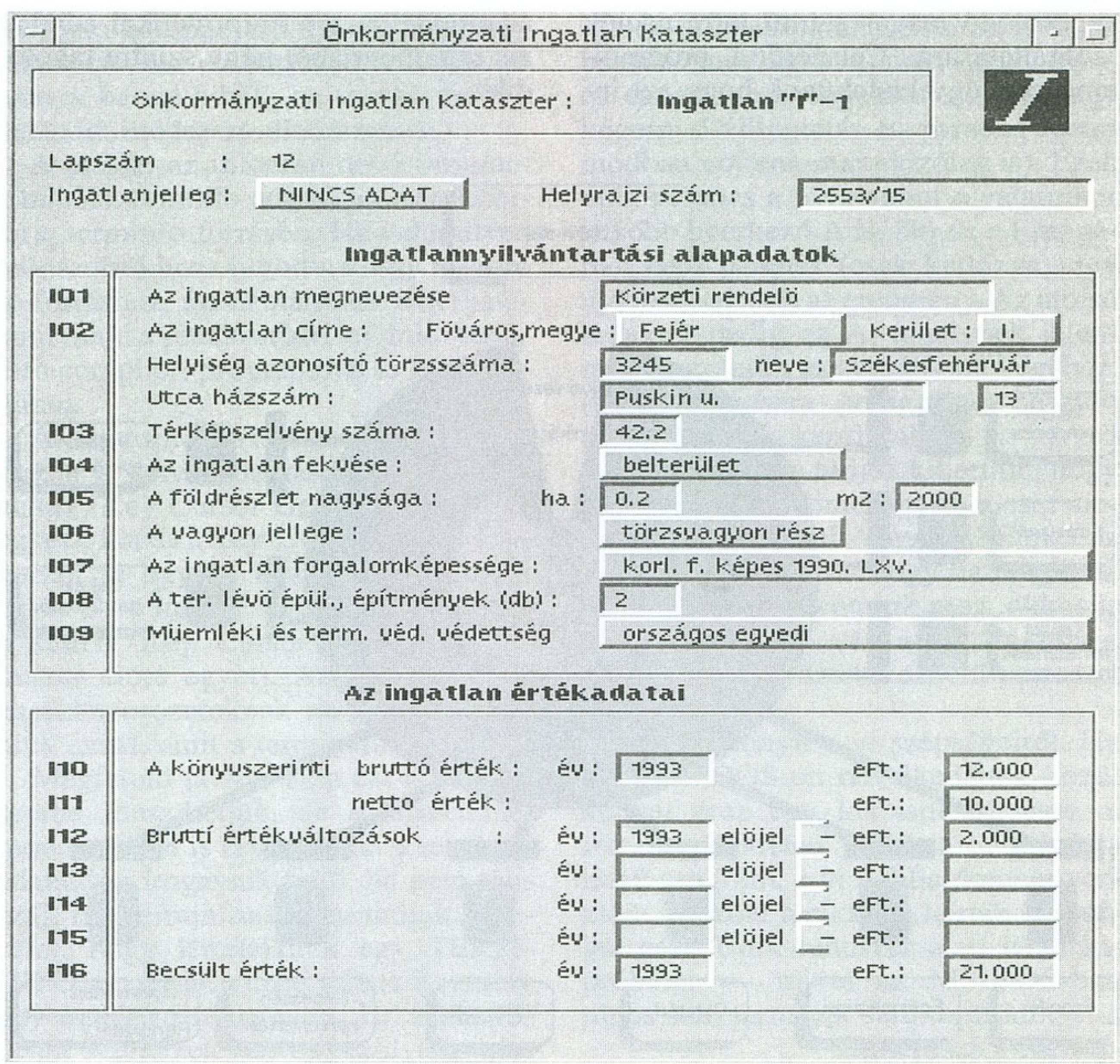
A geometriailag pontos átmeneti fedvényekkel dolgozó alrendszerek egyik példája a területrendezési alrendszer. A területrendezés mint tevékenység olyan átmeneti fedvényeket hoz létre, amelyek az alaprendszer külön rétegen elhelyezkedő, állandó részévé válhatnak — elfogadás és megvalósítás után. Másik példa a lakás és egyéb célú helyiséghasznosítási alrendszer, amely a tevékenység teljes adatkörét és funkcionális feladatait látja el. A szöveges nyilvántartások adattartalmát konvertálás útján lehet a térinformatikai rendszerbe integrálni.

A geometriai pontossági igény nélküli, átmeneti fedvényekkel dolgozó térinformatikai alrendszerek elsősorban ellátástervezési és elemzési feladatokat támogatnak. A rendszerek funkciói a postai cím szintű és aggregált területelemzésre szolgáló alaprendszeri funkciókat használják.

Saját adatbázisaik is ezekre a tematikákra épülnek, így az összerendelés, körzetek kialakítása, elemző statisztikák készítése a külső héjon könnyen megvalósítható. Az alrendszerek térképi geometriai pontossági igénye alacsony, felhasználói felületként Windows alatt futó, grafikus lekérdezőrendszer szolgálhat.

A geometriai igény nélküli alrendszereknek jó példái a szociális, egészségügyi, oktatási, vállalkozói és hasonló alrendszerek, amelyek temporális, historikus fedvényrendszert hozhatnak létre és tartanak karban.

Rudas Pál—Ivanyos János



Hogyan lesz a cserebogár? — II.

A CAA-tól a tökéletes illúzióig

A „varázsló" rendszerek közül néhány — a koraiak többsége — számítógéppel segített animációs rendszer, míg mások modellező animációs rendszerek. A sorozat előző részében vázolt fejlődés (még mindig) korszerű technikáit részletezi a mostani írás.

A számítógéppel segített rendszerek a lehetőségek széles skáláját kínálják. Használatuk egyszerű; az alakzatokat megrajzolhatjuk kézzel, vagy használhatjuk a beépített geometriai szerkesztőfunkciókat (az egyszerűbb rendszerekben pontokat, vonalakat, köröket és téglalapokat rajzolhatunk, míg a bonyolultabb programok sok más lehetőséget is kínálnak). Kivághatjuk a rajz egy részét, majd ecsetként (brush) használhatjuk a rajzolásnál, a képernyő más helyeire másolhatjuk át. Kiszínezhetjük a megrajzolt alakzatokat; különböző színek és mintázatok közül válogathatunk. Megváltoztathatjuk a kép méreteit, vagy akár el is forgathatjuk azt. Háttereket adhatunk a rajzokhoz, és végül elmenthetjük őket.

Számos segédeszköz segíti a tervezést, például különböző interpolációs technikákkal a számítógép kidolgozhatja a kép részleteit, a felhasználónak csak színeznie kell azokat.

A gép képes a köztes animációs fázisok kiszámolására is, ha a rajzok megegyező számú szakaszt és pontot tartalmaznak. Ha két rajz struktúrája nem egyezik meg, a számítógép láthatatlan módosításokat végezhet rajta, melyek csak a komponensek számát változtatják meg, de ezáltal már lehetővé teszik a gépi feldolgozást, valamilyen interpolációs technika alkalmazását. Több ilyen technika létezik, a rajz pontjai mozoghatnak konstans sebességgel, gyorsulva, lassulva, vagy ezeket variálva. Mindazonáltal ha több fázis követi egymást, ezek a mozgások elveszítik a folytonosság látszatát.

A váz és a „töltelék"

Másfajta problémával kerülünk szembe akkor, ha a számítógépnek sok kulcsfázisra van szüksége ahhoz, hogy a mozgás megtartsa tervezett alakját. Napjainkban már több módszer is létezik, mely leküzdi ezeket a problémákat (például a csontváz technikák — angolul skeleton techniques — vagy a P-görbék).

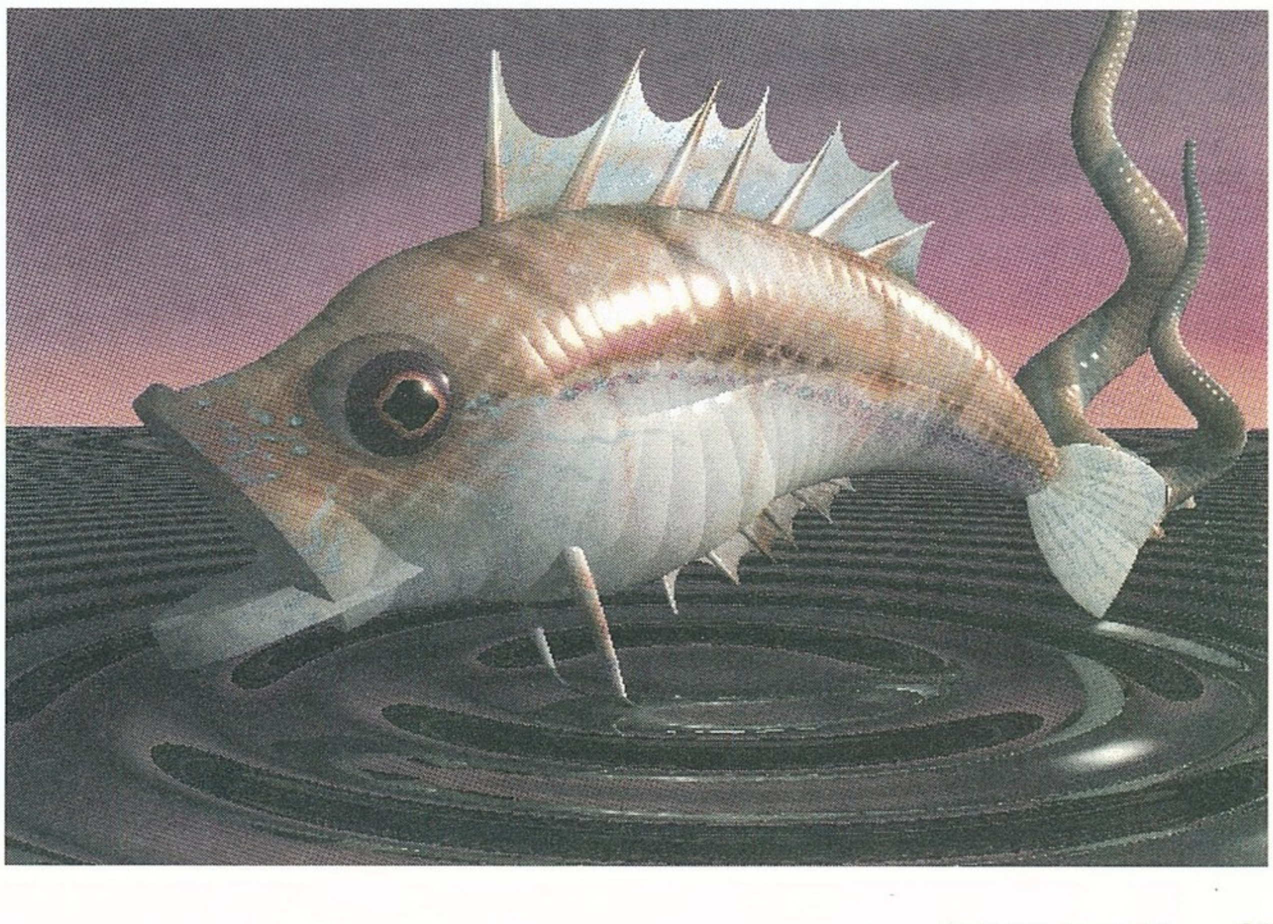
A fázisok elkészítése után az animátor kiszínezheti a rajzokat. Mindössze a kívánt színt és mintázatot, valamint a kitöltendő területet kell megadnia. Ez az eljárás sokkal egyszerűbb és gyorsabb, mint a kézi színezés. A különböző festő rendszerek és grafikus editorok általában kétféle színezést ismernek: egy tetszőleges körbekerített, lezárt terület, vagy egy megadott paraméterekkel rendelkező poligon, illetve egyéb geometriai alakzat (kör, ellipszis stb.) kitöltését. Míg az első módszer képpontokkal (pixelekkel) dolgozik, és általában egy rekurzív algoritmus valamilyen adaptációja, addig a második eljárás ennél jóval komplexebb, a poligon oldalainak helyzetével, vagy a geometriai alakzat valamely más paraméterével dolgozik. Végül létezik néhány olyan gradációs technika, amely fontos szerepet játszik a kétdimenziós grafikus animációban. Az árnyékolás, tükröződések és a fényeffektusok sokkal életszerűbbé tehetik a rajzokat.

A 3D-s „mutatvány"

Természetesen ezek a fejlett technikák szintén megtalálhatóak az új modellező animációs rendszerekben, amelyek már nem csak egyszerű segédeszközök. Rájuk épül a háromdimenziós számítógépes animáció. E rendszerek három legfontosabb felhasználási területe a tárgyak (objektumok) modellezése, a mozgások megtervezése és időzítésük, illetve a képmegjelenítés.

Az objektummodellezés a háromdimenziós alakzatok struktúráját definiálja. Egy tárgy leírásának számos módja van. Felsorolhatjuk például az adott dolog körvonalát alkotó szakaszokat. Ezt dróthálós (vagy angolul wireframe) modellnek nevezzük. Jobb módja a leírásnak a tárgy lapjainak megadása (angolul solid model). Ebben az esetben az objektum felületét kell megadnunk — egy-egy felületi elem megadása akár matematikai kifejezésekkel is történhet.

Hogy a valós világ háromdimenziós tárgyait a fenti logikai struktúrák egyi-





kévé tudjuk leképezni a számítógép memóriájában, három lényegesen különböző módszert használhatunk. Az első a digitalizálás. Ha ezt kívánjuk alkalmazni, először pontokat kell definiálnunk az objektum felületén, majd vagy fotókat kell készítenünk a tárgyról különböző nézőpontokból, és ebből speciális program segítségével rekonstruálnunk az objektumot pontjaiból, vagy valamilyen speciális hardvereszközzel kell e pontok térbeli helyzetét közvetlenül megadnunk.

A második módszer a grafikus editálás. Ez egy felhasználóbarát lehetőség; az animátor közvetlenül láthatja, milyen változtatásokat végzett. Számos szerkesztőrendszer létezik, mellyel a felhasználó felépítheti saját háromdimenziós objektumait. A megtervezett tárgyak floppyn, merevlemezen, vagy más háttértárolón rögzíthetők.

Végül egy objektum generálásának harmadik lehetséges módja a programozás. Az animátor akkor alkalmazhatja ezt a technikát, ha valamilyen komplex tárgyat szeretne modellezni, melynek ismétlődő részletei vannak. Ezek az objektumok könnyen programozhatók. Napjainkban számos grafikus nyelv létezik, melyek egyszerűvé és gyorssá teszik a tervezési eljárást. Ezek a rendszerek képesek mozgások definiálására is: elmozgathatnak (elforgathatnak vagy eltolhatnak) egy tárgyat, megváltoztathatják az alakját, színét vagy méretét, de akár a kamera mozgását is vezérelhetik.

A „koreográfia" és a látványvariáció

Néhány rendszer nem definiálja előre a mozgássorozatokat, ehelyett a fizika törvényeit alkalmazva határozza meg azokat az animátor által tervezett tárgyakra. Így a felhasználó szituációkat alakít ki, és maga sem lehet minden esetben biztos abban, hogy mi fog történni. Ez utóbbi programoknak természetesen több időre és memóriára van szükségük, és még mindig sok hiányossággal rendelkeznek: például nem tudnak alkalmazni néhány alapvető newtoni törvényt.

Az első háromdimenziós rendszerek egyszerű algoritmusokra épültek, de megjelenésükkel felmerült néhány nehéz és fontos, megoldásra váró probléma is, mint például a takart felületek kérdése. Ugyanis mikor egy tárgyat nézünk, annak számos körvonala és felülete takarásban lehet. Ennek megoldására, a látható és láthatatlan felületek meghatározására sok algoritmus született, ezek három nagyobb csoportba oszthatók: a tárgytéren, a képtéren és a prioritáslistán alapulókra.

Egy másik jelentős problémakört alkotnak a fénytörések és tükröződések. Segítségükkel a kép realisztikusabbá, élőbbé tehető. A két legegyszerűbb felületfajta az ideális fényvisszaverő és az ideális fényelnyelő, de a valóságban a felületek legtöbbje e két típus között található. Ezért számos modell született a fénytörések szimulációjára. Ezek összetett módszerek, melyek a fénytükröződéseket alkotóelemeikre bontják. Meg kell még említenünk az árnyékolási és árnyalási technikákat, melyeket akár a tükröződésmodellek segítségével is megalkothatunk.

Fénykövetés, árnyékgenerálás...

Különböző árnyalási modellek léteznek: a konstans árnyalás egy állandó fényerősséget rendel az objektum egy-egy oldalához, míg más eljárások az oldalakon belül is interpolálják az intenzitást. Mindezek az algoritmusok szükségtelenné váltak, amikor a 80-as évek elején feltalálták a fénykövetéses algoritmust, a ray tracinget. Ez egy nagy lépés volt előre.

E technika alkalmazója sokkal életszerűbb 3D képeket készíthet. Az algoritmus a fénynek a forrástól a néző szeméig tartó útjának követésén alapul a töréseken és tükröződéseken keresztül. Az algoritmus egyetlen hátránya a nagy memória- és számítási igény, melyet egy-egy ilyen realisztikus kép megalkotásakor használ.

Végezetül meg kell említenünk néhány más technikát, melyek segítséget nyújthatnak egy sokkal realisztikusabbnak tűnő kép megalkotásában. Ilyenek a felületimintázat-szimuláció, az árnyékgenerálás és a fraktálkészítés. A fraktálokat például partszakaszok, folyók, hegyek, hullámok és más fedőfelületek létrehozására használhatjuk. Ezen algoritmusoknak egy valószerű kép elkészítéséhez már jóval több időre van szükségük.

Szabó Dániel—Ladányi József



WINDOWS PANORÁMA

Amit tudni érdemes!

1994-ben negyedévenként, lemezmelléklettel: Windows Panoráma!

Ha március végéig előfizet, egy szám árát megtakaríthatja!
1460 forint helyett csupán 1095 forint!

A megrendelőlapot
– felbélyegzett borítékban –
az alábbi címre kérjük elküldeni:
Computer Panoráma Kiadói Kft.
Budapest VII.,
Wesselényi u. 17. IV. em.
1077
(Magánszemélyeknek
postautalványt küldünk,
jogi személyek
átutalással is előfizethetnek,
nekik számlát küldünk.)

Elő kívánok fizetni a Windows Panoráma című folyóiratra!

Név:

Postacím:

Bankszámlaszám,
OTP-fiók és alszámlaszám:

(Cégszerű) aláírás:

INFORMÁCIÓKÉRÉS: A0308 ▲

HOLLANDER®

E havi ajánlatunk

PHILIPS MONITOROK	
• 7BM749 (mono), 14", 920x480	*14.500.-*
• 7CM5209, 14", 1024X768	*33.900.-*
• 7CM5279, 14", 1024X768, LR	*36.500.-*
• 4CM4270, 14", 1024X768, LR, NI, Multimédia	*44.500.-*
• 1520 Brilliance, 15", 1024X768, LR, NI, Multimédia	*54.900.-*
• 4CM4770, 17", 1024X768, LR, NI, Multimédia	*97.900.-*
• 4CM6088, Trinitron, 17", 1280X1024, LR, NI	*145.900.-*
• 4CM2799, 20", 1280X1024, LR, NI	*175.900.-*
• C2082, Trinitron, 20", 1280X1024, LR, NI	*285.900.-*
• C2120, Brilliance, 21", 1280X1024, LR, NI	*299.000.-*
PHILIPS CD-ROM	
• CM 206, dupla sebességű, kártyával	*29.900.-*
• CM 215/10, SCSI II., kártyával	*36.900.-*
• CM 215/01, SCSI II., kártya nélkül	*29.900.-*
VERBATIM TERMÉKEK	
• 5,25" HD teflonos, előformatált floppy	*750.-*
• 3,5" HD floppy	*990.-*
• 3,5" HD teflonos, előformatált floppy	*1.290.-*
• DC 2120 cartridge	*1.650.-*
• Optikai és CD-R lemezek	

Viszonteladóknak jelentős kedvezmények! *+ÁFA*

1124 BP., MEREDEK U. 27., T.: 185-3755 FAX: 166-7641
MINTABOLT: 1085 BP., BLAHA L. TÉR 3. T./FAX: 138-4947

INFORMÁCIÓKÉRÉS: A0319 ▲

KÁBELHÁLÓZATOK

PROFON

HELYI KÁBELHÁLÓZATOK
tervezése és kivitelezése

ADATHÁLÓZATOK
– IBM Cabling System
– ETHERNET
– UTP
– Twinaxiális
– Koaxiális
– Egyéb

ERŐSÁRAMÚ HÁLÓZATOK
– Számítástechnikai rendszerekhez

HÍRKÖZLŐ HÁLÓZATOK
– Alközponti hálózatok
– Modemes hálózatok

RACKSZEKRÉNYEK
RACKSZERELVÉNYEK
ÖSSZEKÖTŐ KÁBELEK

1141 Budapest, Egressy út 113/E
Telefon/Telefax: 252-0663

INFORMÁCIÓKÉRÉS: A0335 ▲

Graphicon 1.0

Interaktív függvényvizsgálat

A szoftverportékák bemutatásakor ritkán merészkedünk el a sokszor öncélúnak tűnő, kristálytiszta logikát megtestesítő matematika világába.
Most azonban mégis vállalkozunk egy kis kalandozásra Függvényországban.
Tesszük ezt egyrészt azért, mert jó volt kicsit „nosztalgiázni" az integráltakkal, deriváltakkal, lokális minimumokkal...
Másrészt — s ez a nyomósabb, igazi ok — a külföldi szoftverek dömpingje mellett egyre nagyobb fórumot szeretnénk adni az arra érdemes hazai fejlesztésű programok bemutatásának.

Kicsit faramuci helyzetben vagyunk, amikor azt emlegetjük, hogy a Graphicon magyar program, mert igaz ugyan, hogy szegedi programozók és matematikusok fejlesztették, azonban nem titkolják, hogy elsősorban a nemzetközi szoftverpiacra szánták programjukat. Döbbenten láttuk, hogy a szoftver menürendszere, helpjei, üzenetei és kitűnően felépített kézikönyve — amelyben még a tapasztalt Windows-felhasználók is találnak hasznos ötleteket — angol nyelvű. A „mindent angolul" elv jól tükrözi a magyar piaci viszonyokat...

A fejlesztők állítása szerint, ha megfelelő igény mutatkozik kis hazánkban a Graphiconra, akkor a program következő verzióját már magyar nyelven is közreadják. A többnyelvűség annál is inkább fontos, mert a szoftvert nemcsak — nyelvileg is — jól képzett kutatóknak szánják, hanem oktatási célra is.

Használati utasítás

A program kezelése nem okoz gondot a Windowst ismerő felhasználóknak, mert minden területen igazodik a Microsoft diktálta szabványokhoz. Egy online help és egy 60 oldalas kézikönyv segíti a szoftver tanulását.

A Windows alatt futó program nem igényel különösebben igényes hardverplatformot, elvileg már elindul egy 286-os gépen is, ha azon működik a Windows. Igazán jól azonban azokon a gépeken érzi magát a program, ahol — a magas számítási igényhez — van koprocesszor is. Megnéztük a Graphicont egy „fapados" 386-oson, majd egy matematikai segédprocesszorral ellátott 386-oson is, s a sebességbeli különbség érezhető volt... De a program legjobb formáját egy 486-os, koprocesszoros gépen futotta.

A szoftvernek nem nagy a helyigénye. Az installálásához 1,5 Mbájt szabad lemezterületre van szükség, utána már csak 800 kbájtot foglal le a winchesteren. A függvényfájlok mérete sem nagy, 2-3 kbájt. Ezek után nézzük, mit is tud a Graphicon!

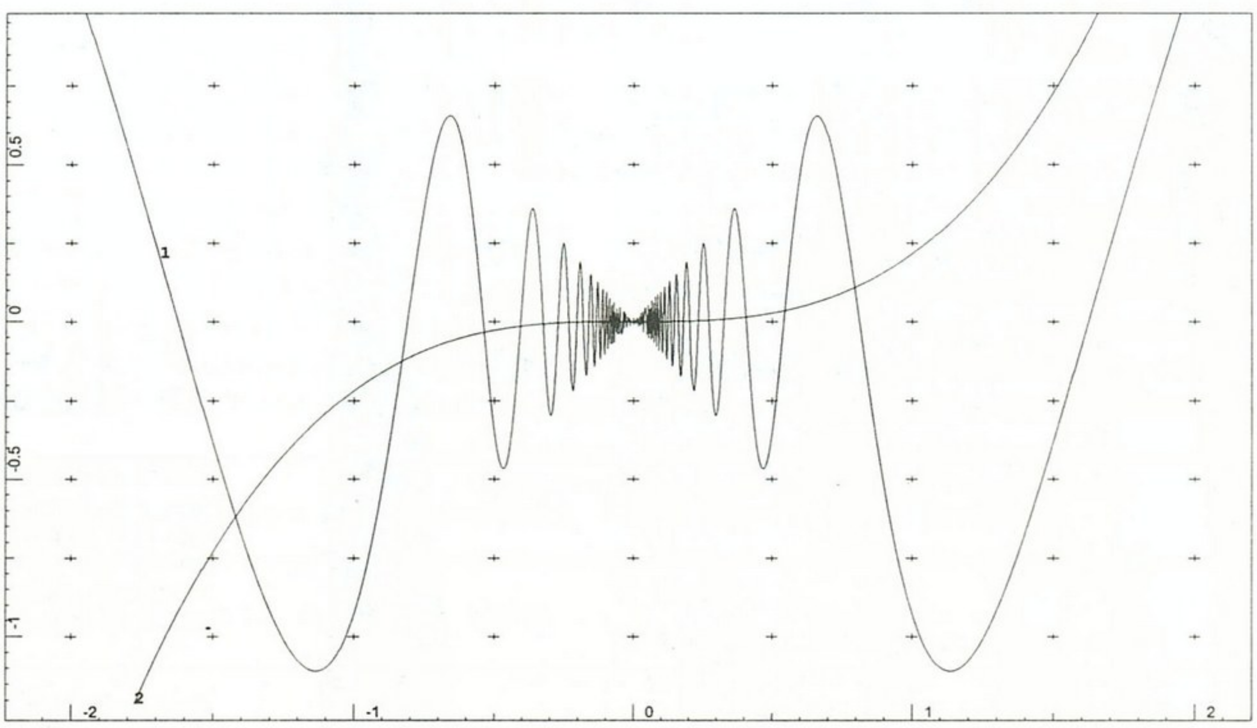
Függvénytanulmányok

Ami a legfontosabb: tetszőleges egyváltozós, paraméterezett függvények — $f(x) = a\sin(bx)+c\cos(dx)$ — jeleníthetők meg egy ablakban, és interaktív módon tanulmányozható azok viselkedése. Így például az órán a tanár sokkal egyszerűbben, gyorsabban és szemléletesebben taníthatja meg a függvények tulajdonságait és a függvénytranszformációk hatásait, mintha a táblán rajzolgatna. A Graphiconnal könnyen módosíthatja (billentyűzetről, egérrel) a függvények paramétereit, ennek eredményét a diákok azonnal látják a függvénygörbe alakján. A függvény paramétereinek változtatásával (egy függvényformula maximum 15 változtatható paramétert tartalmazhat) jól megfigyelhető a függvények viselkedése is,

Sample

```
Function name    : Sample
Filename         : D:\CYGRON\GRAPHICN\CPFUN\CP07P.FUN
Function formula: f(x)=f*(x*sin(a/(b*x)))+g*((h*x)^3)


X axis interval : [-2.1697169717 , +2.20372037204]
Y axis interval : [-1.21262331188 , +0.979185232115]     Created by Graphicon (14 Nov, 1993 - 11:35)
```




```
Parameter settings:
No.        A        B        F        G        H
 1.       5.1       1        1        0        0
 2.       3.3      0.8       0        1       0.6
```

és a különböző (maximum 10) paraméterbeállítású függvények, görbék össze is hasonlíthatók. A függvények görbéi megjeleníthetők egy ablakban, így egyenletek és kétismeretlenes egyenletrendszerek vizuálisan megoldhatók.

Többféle segédeszköz (rács, szálkereszt, nagyítás, színbeállítások és egy dinamikusan változó státus-sor) támogatja a függvények vizsgálatát.

A paraméterbeállításokat függvényduplázással megőrizhetjük, és kijelölhetjük azokat az aktív függvényeket, amelyekre a paraméterváltoztatások és egyéb függvényműveletek vonatkoznak. Természetesen a függvénygörbéket kinyomtathatjuk, vagy clipboardba másolhatjuk.

Függvénykapcsolatok

A függvényekkel különböző számításokat is végezhetünk. Az ehhez szükséges intervallumokat két jelzővonal segítségével egérrel adjuk meg, majd az adott intervallumon kiszámíthatjuk a függvény integráljának értékét, megkereshetjük a lokális minimumát és maximumát, és egy pontban meghatározhatjuk a függvény első és második deriváltjának értékét.

A Graphicon másik fontos felhasználási területe: az adatok közötti függvénykapcsolatok keresése. Például az üzletemberek az adatok alapján tendenciákat állapíthatnak meg, és következtetéseket vonhatnak le a valutaárfolyamok vagy a termékárak alakulására, és elemezhetik a tőzsdei adatokat; a kutatók pedig analizálhatják elemzendő mintájuk viselkedését.

A grafikonon adatpontokat (legfeljebb 500-at) is megjeleníthetünk. A szoftver méri a négyzetes eltérést az adatpontok és a függvény között, és a függvényt egy lokális optimalizáló algoritmus (Fletcher–Powel) az adatpontokra illeszti.

Az adatpontok koordinátáit megadhatjuk a Graphiconon belül, de beolvashatjuk különböző adatfájlokból is, így például Lotusból, Excelből, dBase-ből vagy szövegfájlokból. A Graphiconban minden adatot külön fájlba menthetünk, ezeknek címkét is adhatunk, így azok gyorsan visszakereshetők.

A Cygron Kft által fejlesztett és forgalmazott szoftver elindult a remélt nemzetközi karrier útján, máris büszkélkedhet olasz referenciákkal. De felfigyeltek a szoftverre a magyar kereskedők is, és a szoftver demóváltozatát megjelentették a SolarSoft kínálatában.

Magyar ár, külföldi ár

Bár sokak szerint nem érdemes (sőt egyenesen tilos!) megvenni egy szoftver 1.0-ás verzióját, ezt a programot „anglomániája" ellenére mégis bátran ajánljuk iskoláknak, kutatóknak egyaránt. A fejlesztők ígérik, hogy hamarosan kirukkolnak egy következő verzióval, amelyben a kétváltozós függvények háromdimenziós ábrázolását valósítják meg.

S végül egy pozitívumot még mindenképpen ki kell emelni a szoftverrel kapcsolatban: olcsó. Sőt! A legális hazai szoftverkereskedelem elősegítése érdekében a magyarországi Graphicon-árak alacsonyabbak, mint a külföldiek. 20 ezer Ft a hazai ár, de oktatási intézmények féláron juthatnak a szoftverhez. Ugyanez külföldön 300, illetve 150 dollárba kerül.

Sziebig Andrea

New kids in the block, PowerPC

Kivel van az erő?

Rovatunk processzorológiai sorozatát egy kis kitérő kedvéért megszakítjuk. Mivel ez a kitérő a PowerPC, úgy véljük, sokan lesznek, akik nem neheztelnek meg érte. A PowerPC nem számítógépet, hanem processzorcsaládot jelöl, amely az IBM, az Apple és a Motorola közös fejlesztésének az eredménye. A család első tagja, a 601-es 1993 nyárutóján debütált, az év végére pedig megjelentek a vele készült első gépek. Mivel az iparág történetében ritka az ilyen — a nyilvánvalóan versenytársak közötti — együttműködés, érdemes megvizsgálni a körülményeket és a várható hatásokat. Magát a processzort a Mikroprocesszorok minienciklopédiája sorozatban mutatjuk majd be.

Az IBM, az Apple és a Motorola képviselői által 1991. július 3-án aláírt szerződésben nemcsak az új processzortípus kifejlesztéséről, de az ezen futó operációs rendszerről és a szoftvertámogatásról is megállapodtak. Az IBM súlya meghaladja két partneréét, így valószínűleg ő választotta ki a társakat, mintsem fordítva.

Az előzmények formálták a célokat

A Motorola részvételét az indokolja, hogy a 4. legnagyobb félvezetőgyártó, és mind a CISC (68000-es sorozat), mind a RISC (8800-as család) CPU-k gyártásában vannak tapasztalatai. Az Apple pedig azon ritka cégek közé tartozik, amelyek a hardvertervezéstől az operációs rendszeren át az alkalmazói szoftverekig a teljes kínálatot maguk készítik.

Az új architektúra alapját az IBM sikeres RISC-alapú munkaállomása, az RS/6000 képezi — ebben a központi egység 5 db integrált áramkörből áll. Ezek funkcióját egyesíti egyetlen chipen a PowerPC, amely binárisan kompatibilis az RS/6000-rel. A legfontosabb célkitűzés az volt, hogy a PowerPC széles teljesítménytartományt tudjon átfogni, így a hordozható gépektől a hálózati szerverekig és munkaállomásokig egymással binárisan kompatibilis gépek legyenek építhetők vele. A család a következő típusokat tartalmazza:

MPC601: 32 bites processzor, 32 bites címbusszal, és 64 bites külső adatbusszal, a busz szerkezetileg a Motorola 88110-es RISC típusán alapul. Viszonylag olcsó asztali gépekbe szánják. Megjelent: 1993 augusztusában.

MPC603: Jellemzőiben megegyezik a 601-gyel, de alacsonyabb fogyasztású, statikus működésű (azaz az órajele működés közben megállítható), így elsősorban hordozható gépekben használják majd. Tervezett megjelenés: 1994 közepe.

MPC604: Jellemzőiben megegyezik a 601-gyel, de belső struktúrája nagyobb teljesítmény elérését teszi lehetővé, gyors asztali gépekhez készül. Tervezett megjelenés: 1994 közepe.

MPC620: 64 bites cím- és adatbusz, binárisan kompatibilis a család többi tagjával, de lényegesen nagyobb teljesítményű, gyors munkaállomások és hálózati szerverek számára. Tervezett megjelenés: 1994 vége.

Az IBM és a Motorola közös tervezőközpontot hozott létre a texasi Austinban. Az 50 és 66 MHz-es változatban elkészült 601-es teljesítménye nagyjából megegyezik az azonos órajelű Pentiuméval (lásd a táblázatot), ára azonban lényegesen alacsonyabb. A hármak szándéka szerint a PowerPC-vel olyan olcsó gépet vehet a majdani felhasználó, amely nemcsak gyors Unix munkaállomás, de használhatóak rajta a könnyen kezelhető Macintosh, és rendkívül széles választékú DOS/MS-Windows programok. Mindezek birtokában pedig képes versenyre kelni a személyi számítógépek piacát szinte kizárólagosan uraló IBM PC/AT-vel.

Miért RISC?

A nyolcvanas évek második felében kezdték alkalmazni azokat a processzortervezési alapelveket, amelyeket ma gyűjtőnéven RISC (Reduced Instruction Set Computer) technológiának nevezünk. Az Intelnek a PC-ben alkalmazott x86-os, illetve a Motorolának a Macintosh gépekben használt 680x0 sorozatának indulásakor ezek még nem voltak ismertek, ezért ezeket a típusokat CISC (Complex Instruction Set Computer) névvel jelölik. A RISC-elv sikeresnek bizonyult, az így épített CPU-k teljesítménye messze meghaladta a CISC típusokét. Ezek mögött azonban az eltelt évek alatt hatalmas szoftverbázis nőtt, és a tömeggyártás folytán áruk olyan szintre csökkent,

A PowerPC és a Pentium összehasonlítása (1994 eleji ár, 1000 db-os tétel esetén)

Típus	Sebesség	Cím	Külső adatbusz	Belső cache	Spint 92	Spfp 92	Méret	Max. fogyasztás	Tranzisztorok	Ár
MPC601	66 MHz	32 bit	64 bit	32 K (egységes)	60	80	120 mm^2	9 W	2,8 millió db	465 dollár
Pentium	66 MHz	32 bit	64 bit	8 K adat, 8 K utasítás	64,5	57	292 mm^2	16 W	3,1 millió db	871 dollár

hogy a korábbi számítógépeknél elképzelhetetlen mennyiségben keltek el. Becslések szerint több mint 140 millió IBM PC-t használnak a világon. A RISC processzorokat alkalmazó, a PC-nél jóval nagyobb teljesítményű munkaállomások ára is csökkent, de piaci részesedésük — persze az eladott darabszámot, és nem a forgalom értékét tekintve — a PC-hez képest elenyésző.

A RISC-elveket részlegesen az Intel is alkalmazza, a Pentium sokak szerint egyfajta RISC-CISC keveréknek tekinthető, de a tervezők kezét köti a korábbi típusokkal való kompatibilitás követelménye, ezért nem érik el a tisztán RISC típusok sebességét.

A három cég szerint a PowerPC-vel eljött az ideje, hogy RISC processzort használjunk a hétköznapi számítástechnikában is. Az alacsony ár érdekében egy már létező RISC architektúrát vettek alapul, és a fejlesztési költségeket megosztották egymás között.

A szoftvereket „nem kell félteni"

Az RS/6000-esek AIX operációs rendszere (az IBM Unix-változata) és minden programja a bináris kompatibilitás miatt változtatás nélkül tehető rá a PowerPC-vel működő gépekre, de megegyeztek egy új operációs rendszer, a Unix OSF/1 szabványon alapuló PowerOpen kifejlesztésében is.

A korábbi, nem PowerPC-re és RS/6000-re készült szoftverek használatát emulációval tervezik megoldani, amíg az új szoftverháttér ki nem alakul. Így nem kell eldobni a már megvásárolt programokat, másrészt áthidalható az új hardver bevezetésekor jellemző szoftverhiány.

Különböző emulációs módszerek léteznek. Közös jellemzőjük, hogy alkalmazásuk elég nagy teljesítményveszteséget jelent az adott processzorra fordított, annak saját gépi kódján, „bennszülött" (natív) módon futó programhoz képest. Például az Insignia Solutions SoftPC 3.0 nevű MS-Windows 3.1 és DOS 5.0 emulációja a Mips R4000-es RISC processzorán fut. Vele egy jó 386-os gép előtt érezhetjük magunkat, noha az R4000 teljesítménye kb. 8-szorosa a 33 MHz-es 386DX teljesítményének.

A PowerOpenbe beépítik a DOS/MS-Windows és a Macintosh operációs rendszerének a rendszerhívásait, amelyek már natív módon futnak. A grafikus felületen futó alkalmazói programok futási idejük 60-90%-ában a rendszerhívásokat használják, ezért remélhető, hogy az emuláció teljesítményvesztesége nem lesz olyan nagy, mint a korábban említett példában. A PowerOpent — saját fejlesztéseivel kiegészítve — az IBM és az Apple is kínálja majd.

A versenytársak közötti megállapodás nagy szenzáció volt, de az eredményben sokan kételkedtek, mivel annak idején mindhárom cég kijelentette, hogy változatlanul folytatja korábbi — a PowerPC-vel értelemszerűen konkuráló — fejlesztéseit is. A család első processzorának tavalyi megjelenése azonban eloszlatta az aggályokat, és növelte az érdeklődést az architektúra iránt. Eredetileg csak a DOS programok futtathatóságáról volt szó, a Windows-támogatást később határozták el. Az Insignia Solutions a már említett SoftPC-t elkészíti a PowerPC-re is.

Az IBM az OS/2 operációs rendszert is kínálja majd PowerPC-hez, az Apple pedig a System 7-et fogja (nem csak emuláció formájában) elérhetővé tenni.

A 601-est leggyakrabban a Pentiummal hasonlítják össze, mivel elsősorban ennek a típusnak jelent konkurenciát: a 601-gyel épített olcsó gépek a pentiumos PC-kkel szemben jelentenek alternatívát. Ez azonban valószínűleg nem ingatja meg az Intel pozícióját.

A cég jelenleg a legnagyobb félvezetőgyártó a világon, processzorgyártó kapacitása pedig többszörösen meghaladja a versenytársakét. Elsősorban a PC-kben alkalmazott CPU-ról ismert, de gyárt RISC típust is, a 32 bites 860-ast. Az 860-nal épített párhuzamos gépei nemrég megelőzték a korábban vezető CRAY szuperszámítógépeit.

A PowerPC sikere esetén az Intel egy éven belül képes kihozni egy vele kompatibilis típust, vagy ha olcsóbb, megvásárolja a licencet, mivel a hármak a PowerPC-t nyitott architektúrának szánják. A mellékelt táblázat adatai a Pentium ellen szólnak, de ebben a típusban még jókora tartalékok vannak. A technológia fejlesztésével csökkenthető mind a fogyasztás, mind a költségeket erősen befolyásoló méret. A Pentium 94-ben megjelenő új változata 100 MHz-et vagy ennél nagyobb belső órajelet használ majd. Az Intel 93 végén csökkentette árait — többek szerint ez is a PowerPC hatása: a Pentiumét 3%-kal, a legnagyobb mértékben pedig a 486DX2-50 MHz-esét (10%-kal). 1994 második negyedévére pedig a Pentiumnál 14, a 486DX2-66-nál 18%-os csökkentést jelentett be.

Az új processzorra az Apple-nek van leginkább szüksége, a cég ugyanis most komoly bajban van. A Macintosh annak idején két évvel előzte meg az IBM

PC-jét, de ennél is nagyobb előnye volt abban, hogy a hardvert és a szoftvert egységben, egymáshoz illeszkedve tervezték és készítették. A PC-t az IBM eredetileg csupán egy intelligens terminálnak szánta, nem fektetett energiát a szoftverfejlesztésbe, az operációs rendszert is egy külső cégre (a Microsoftra) bízta. A hardver- és szoftverfejlesztés összehangolatlansága évekig meglátszott a PC-szoftverek minőségén.

Az almába beleharaptak...

Az Apple nem adta ki kezéből a gyártás jogát, és ez az üzletpolitika sokáig sikeres volt, szavatolta a hardver megbízható minőségét. A PC-k azonban ezalatt példátlan áresésen, és — az újabb Intel processzoroknak köszönhetően — jókora teljesítménynövekedésen mentek keresztül. A teljesítménynövekedés pedig olyan szoftverek, elsősorban a grafikus felhasználói interfész (MS-Windows) használatát teszi lehetővé, amilyenek korábban csak a Macintosh-on voltak elérhetőek. Megindult a jellegzetes Macintosh-programok átvándorlása a PC — illetve az MS-Windows — alá, az olcsó PC így egyre versenyképesebb alternatíva lett az Apple gépeivel szemben.

Ráadásul a Macintoshok processzorát készítő Motorola egyre késik az új típus gyártásával. Az Intel és a Motorola évekig fej-fej mellett haladt a fejlesztésben, így 1989-ben néhány hónap eltéréssel jelentették be a 486-ot és a 68040-et.

A Pentium tavaly márciusi megjelenésére a Motorola mindmáig nem válaszolt. A 68060-at (a 68050-es típusszámot a hírek szerint kihagyják) csak az idei év közepére ígérik, és az előrejelzés szerint teljesítményben elmarad a Pentiumtól, az 50 MHz-es első változat 70-80 MIPS-et tud majd.

Az Apple 93 őszén 16 ezer alkalmazottjából 2500-nak mondott fel, és befagyasztotta a fizetéseket. Gőzerővel dolgoznak a PowerPC-n alapuló új gépeken, és az emuláció révén zökkenőmentes átállást ígérnek a szoftverek területén. Közkedvelt szoftvereit az emuláció korlátai miatt a PowerPC-re is elkészíti majd, ekkor lesz teljesen kihasználható az új típus teljesítménye.

A PowerPC-n alapuló Macintosh a tervek szerint 1994 első felében jelenik meg.

A PowerPC-ből valószínűleg az IBM profitál a legtöbbet. A személyi számítógépek világában ez rá is fér, mert — noha ő az eredeti PC, XT és AT szülőatyja — ez a gépkategória az utóbbi években nem sok sikert hozott számára. Rosszul mérték fel a várható fogadtatást, majd a PS/2 architektúra szabadalmi védelmével próbálták meg távoltartani a klóngyártókat. Ezek a licencdíj fizetése helyett az újabb processzorokkal tovább gyártották az eredeti AT-t (amivel az IBM a PS/2 megjelenésekor felhagyott), és létrehozták az EISA-szabványt.

A PowerPC elsősorban azt jelenti az IBM számára, hogy saját RS/6000-es architektúrája — az Apple és a Motorola teljes támogatásával — a PC-vel versenyképes áron is megjelenik. Valamennyi, az RS/6000-re kifejlesztett szoftver használható a PowerPC-n, a gép megjelenésekor ő kínálja a legteljesebb szoftverválasztékot.

Az IBM maga is gyártja a processzorokat, így az Apple-lel ellentétben független lehet a Motorolától. Így például a nemrég megjelent RS/6000 típusú 250-es gépébe már az új processzort építette be.

A megállapodás és az új CPU-család értékelésekor ritkán kerül szóba a Microsoft neve, pedig ez hosszú távon az ő számára jelenthet kihívást.

A Microsoft a legnagyobb szoftvergyártó, nagyságát a PC elterjedésének köszönheti. A PC népszerűségének esetleges csökkenésével az ő részesedése is automatikusan csökkenne. Ennek természetesen a cég vezetése is tudatában van, a Windows NT operációs rendszer eleve háromféle (Intel, Mips R4000 és DEC Alpha) processzortípusra való bejelentése is mutatja, hogy meg akarja szüntetni az egy lábon állást. Programjai azonban a PowerPC-n az első időkben csak emulációval lesznek használhatóak, és ennek hatásfoka egyelőre bizonytalan.

A PowerPC-vel legtöbbet nyilván a felhasználó nyer majd, aki az IBM PC-n használható DOS/MS-Windows, Unix, OS/2 és NextStep mellé a PowerPC-n az Apple System 7 operációs rendszerét is futtathatja a Macintosh programokkal együtt.

Az, hogy az új architektúra versenytársak közötti megállapodás alapján született meg, azt is jelezheti egyben, hogy a cégek rájöttek: a kíméletlen verseny végén valamennyien vesztesek. A versenynek és az ezzel járó árcsökkenésnek persze minden felhasználó örül, de nem szabad elfelejteni, hogy a jelenlegi profit finanszírozza a jövőben használatos eszközök fejlesztéseit. Ha a haszon túl alacsony, a cég nem kezd új fejlesztésbe, ami végső soron senkinek sem érdeke.

Csórián Sándor

Haladjunk a korral! — I.

Levelezés elektronikusan

Ebben a cikksorozatban elsősorban azokat a lehetőségeket ismerhetik meg olvasóink, amelyeket az elektronikus levél (E-mail) révén használhatnak ki.
Bizonyos funkcióihoz Magyarországon jelenleg még nem lehet hozzáférni, de ami késik, nem múlik.

Elég rég megfogalmazódott az igény, hogy az egyik gépen nyert eredményeket, adatokat egy másik gépen is használhassuk. Ilyenkor az a gyakorlat, hogy valamilyen adathordozóra (jelenleg mágnesszalagra vagy mágneslemezre, régebben lyukszalagra vagy lyukkártyára) rögzíti az ember az adatait, s maga viszi át a másik géphez, vagy valaki más (nagyon nagy távolság esetén a posta) gondjaira bízza. Felmerült a kérdés, lehet-e másképpen? Mint az írott levélhez képest a továbblépést a telefon jelentette, a számítógépnek a hálózat a megoldás. Ugyanúgy kell elképzelni, mint a telefont. Egyetlen különbség, hogy itt nem az emberi hangot alakítjuk át elektromos jellé s vissza, hanem a számítógépek bitjeit. A hálózatok kábelei általában speciális telefonvonalak, amelyek szünet nélkül üzemelnek.

Hálózatban a világgal

Az első szervezet, amely először kapcsolt össze egymástól nagy távolságra lévő gépeket, az USA Hadügyminisztériuma volt, még 1969-ben. Ennek a hálózatnak nagyon nagy sikere lett, így az amerikai egyetemek is elkezdték saját hálózatukat építgetni. Ezek után már nem volt megállás: a kisebb-nagyobb cégek sem akartak lemaradni. Lassan már a háztartásokba is kezd behatolni ez a „közeg".

A legfontosabb lépés az volt, hogy ezeket a hálózatokat Internet néven egységes hálózatba kapcsolták. Az Internetbe kötött gépek száma jóval meghaladja az egymilliót, s egy-egy ilyen gépet nem csak egy ember használ. Így el lehet képzelni, hány és hány „társsal" kerülhetünk kapcsolatba, s mennyi tudás, ismeret halmozódott fel, amit könnyedén elérhetünk.

Ha valakinek van valamilyen problémája, csak ír egy levelet, amelyre órákon, esetleg napokon belül ott a kimerítő válasz. Más: ha korábban valamiért kellett egy program számomra, megpróbáltam megírni. Ha ma kell valami, csak írok egy levelet, s az egyik ezzel foglalkozó gép körbenéz, megtalálható-e valahol? Ha igen, egy másik levél hatására már meg is kaptam a programot levélben. Ha nincs meg az a program, csak akad valami hasonló (amelyet ugyanígy megszerezhetek!), s annak alapján sokkal könnyebb megírni a számomra szükséges programot. Sőt, olyan dolgokat is meg lehet ismerni, amelyekről a hálózat nélkül nem is hallott volna esetleg az ember.

Számokkal és jelekkel

Minden Internet-beli gépnek van egy 32 bites egyedi azonosító száma. Ezt a számot bájtonként használjuk, de mivel a kettes számrendszer elég kényelmetlen, tízes számrendszerben. Hogy érhetőbb legyen: annak a gépnek, amellyel dolgozom, 193.6.128.230 a száma. Egy ilyen számot nehéz fejben tartani, főleg ha az ember 20-30 címmel dolgozik nap mint nap. De néha az is előfordul, hogy a gépet kicserélik egy jobbra, s a szám is megváltozik. Most erről mindenkit értesítsünk vagy senkit? Egyik sem jó megoldás. Épp ezért minden gépnek van egy neve, s majdnem mindig ezt a nevet használjuk a szám helyett, ha a gépre hivatkozunk. (A gép megváltozhat, de a név általában megmarad.)

Az előbb említett gépnek a neve {llab.arts.klte.hu}. Ha valaki nagyon ért a rövidítésekhez, akkor kitalálhatja, mit jelent ez a név, a többieknek elmagyarázom: az {llab} a language laboratory rövidítése, ugyanis ezt a gépet főleg nyelvész diákok nyúzzák. Az {arts} a Bölcsészkar, a {klte} a Kossuth Lajos Tudományegyetem, míg a {hu} a Hungary rövidítése. Mint az előbbi címnél, az Internet-címek nagy része az országra utaló taggal ér véget. Az USA már megint kivétel, itt több különböző végződés terjedt el. A {mil} a Hadügyhöz tartozó, az {edu} az egyetemekhez, a {gov} a közigazgatási intézményekhez, a {com} a vállalatokhoz, a {net} a különböző csatolt hálózatokhoz, az {org} a különböző szervezetekhez tartozó gépek címének utolsó tagja.

Hazánkban az Interneten kívül egy másik hálózat is elterjedt, a Bitnet (Because-It's-Time Network mozaikszava). Ez elsősorban oktatási intézményeket köt össze. Itt is minden gépnek egyedi neve van. Például az egyetemünkön lévő gép címe {huklte51}. Ha az Internetről akarunk bitnetes címre levelet írni, a gép neve után kell illeszteni a {bitnet} szócskát: {huklte51.bitnet}. A két hálózat között sok kapcsolódási pont van, sok gépet meg lehet címezni mind internetes, mind bitnetes módon is.

Az Internet a TCP-IP szabványt használja. A TCP felelős az adatok csomagokra bontásáért, az IP pedig a csomagok továbbításáért. A korábban említett számokat ezért is hívjuk IP-számnak. Ha hosszabb levelet küldünk, az több részre bomlik, és külön-külön továbbítódik. Nem lehet csodálkozni, ha egy korábbi rész később ér a címzetthez. A fogadó gép megvárja az összes részt, s összerakja a helyes sorrendben. Csak akkor kaptuk meg a levelet, ha az összes része megérkezett. A hálózaton ezek a részek a {tárold és küldd tovább} elven továbbítódnak, azaz ha megérkezik egy rész helyesen, azt az adott gép tárolja, és továbbküldi a következő csomópontnak, s amíg a következő nem jelzi vissza, hogy hibátlanul megkapta, addig nem törli.

Miért előnyös az elektronikus levél?

A legfontosabb előny a gyorsaság. Percek alatt elér a föld bármely sarkába. Nem kell heteket várni a válaszra. Ha kaptunk egy levelet, nem kell bajlódni a küldő kézírásának kisilabizálásával. S ha a számítógép ott az ember asztalán,

nem kell elsétálni a legközelebbi postaládáig. Nem kerül pénzbe! (Ez az, amit nem képesek elhinni az emberek, pedig majdnem így van. Nálunk az egyetem Budapest és Debrecen között bérel egy telefonvonalat, amelynek a használatáért átalánydíjat fizet, de ezt nem hárítja az oktatókra, diákokra. Az USA-ban vannak olyan telefonszámok, amelyeket ha felhív az ember, és modemmel rákapcsolja a számítógépét a telefonra, élvezheti a hálózat nyújtotta szolgáltatásokat. Persze a telefonszámlát fizetni kell!)

Szólhatunk azért hátrányokról is. Nincs meg az az érzés, mint amikor az ember a borítékot nyitja fel. Baj van a nem szokásos karakterek küldésével (akárcsak a táviratban): az ékezetes betűkkel, a cirill betűkkel. Akárcsak a normális levelet, az elektronikus levelet is elolvashatja a címzetten kívül más is. De mivel igazán összekeverni valamit csak számítógéppel lehet, így könnyedén lehet leveleinket titkosítani, s úgy továbbküldeni. Valamint ami a legveszélyesebb: az ember a levelek rabja lehet. Könnyedén kaphat az ember napi több száz levelet, s azokat mind feldolgozni képtelenség.

Térjünk vissza egy kicsit az ékezetes betűkhöz! Ha az ember leül egy gyors levelet írni, nem fog az ékezetekkel törődni. Ha fontosak az ékezetek, több elterjedt kódolás is rendelkezésre áll már — természetesen a küldőnek és a fogadónak is ugyanazt kell használnia. S az elektronikus levélben mindent el lehet küldeni, amit át lehet alakítani számítógép által elfogadható adattá. Ezért nem csoda, hogy vannak már olyan levelezőrendszerek, amelyekben hangot és képet is lehet küldeni. De mi ezzel most nem foglalkozunk. Nincs egységes szabvány a levelezőprogramok felületére. (Ahányfajta gép, annyi program, sőt!) Vannak ablakos, egérrel kezelhető felületek, s bizony vannak ezekhez képest fapadosak is. Most két ilyenről lesz szó: az egyik a Sun Unix, a másik a VAX VMS alatti levelezőrendszer.

A minimum

Csak a lehető legkevesebbet fogjuk bemutatni, amit minden levelezőrendszernek tudnia kell. Ezek a következők: levelet írni, küldeni, fogadni, elolvasni, levélként elküldeni egy már kész fájlt, s levelet fájlba kimenteni. Nagyon sok további kényelmes lehetőség van, de ezek erősen rendszertől függőek; javaslom, ezekről kérdezze meg mindenki a saját rendszergazdáját.

Mindenki, akinek accountja van, rendelkezik egy login névvel. Nálam ez az {aszalos}. A cím a login névből és a gép IP-nevéből áll össze. Az én címem ezek alapján:

{aszalos@llab.arts.klte.hu}.

Ha egy levelet akarunk küldeni, akkor mindenképpen ismerni kell a címzett címét, PONTOSAN! Elég egy betűt eltéveszteni, s a levél már ismeretlen címzett jelzéssel jön vissza. Ha megnézünk egy levelet (a lemezmellékletben sun.lev, illetve vax.lev), a következőket figyeljük meg: {From}, {To}, {Subject}, {Date/Time}. Vegyük sorra:

{From} — Ez jelzi, ki írta a levelet. Néhány gépnél valamilyen okból ez hibás. Így van ez nálam is. Nagyon kellemetlen, mert sok szolgáltatás az itt található címre akarja a választ küldeni.

{To} — Itt a címzett címe szerepel ({loginnév@@gépnév}). Ha a címzettnek is ugyanazon a gépen van accountja, mint nekünk, akkor a gépnév elhagyható. (Ez az, amit a küldőnek mindenképp ki kell töltenie.)

{Subject} — Ez a sor annak jelzésére szolgál, miről is szól a levél. Ha több tucat levelünk érkezett, ez alapján a fontosabbakat kiválogathatjuk.

{Date/Time} — A levelezőprogramok automatikusan megjelölik, mikor küldték a levelet.

Lássuk a lényeget!

A levelezőprogram szövegszerkesztője általában igen egyszerű, keveset tud. Lehetőleg ne írjunk 70 karakternél hosszabb sorokat, s mivel ha a sort lezártuk, akkor nehéz visszamenni és kijavítani, ezért óvatosan gépeljünk. Ha hosszabb levelet akarunk írni, javaslom: mindenki írja meg kedvenc szövegszerkesztőjében, s mentse ki ASCII formátumban, mert egyébként az átvitel eredménye kétséges.

Vegyük először a unixos levelezőprogramot. Természetesen jelentkezzünk be. A program neve {mail}. (Mi más is lenne?) Ha levelet akarunk küldeni, akkor a következőket gépeljük be: '{mail} a_címzett_címe'. A ' jelet az idézőjel helyett használom, csak e jelek közti részt kell értelemszerűen begépelni, így most a_címzett_címét helyettesítsük valamilyen címmel, például a saját login-nevünkkel. Ne felejtsük el se a {mail} utáni szóközt, se a sort záró billentyűt (CR, Return vagy Enter). Ezek után a gép a {Subject}-et kéri. Ide tetszőlegesen írhatunk bármit. Ezt a sort is, mint majd a többieket, zárjuk Enterrel (vagy a megfelelő billentyűvel). Most szabadon megírhatjuk a levelet. A levelet úgy küldhetjük el, hogy az utolsó sorába csak egy pontot teszünk. Ezzel a levelet el is küldtük. Próbaképpen érdemes írnunk egy levelet — magunknak!

Ha egy fájlt be akarunk illeszteni a levélbe, a következőképpen tehetjük meg: A sor elejére a következőket írjuk: '{~r} fájlnév' ezzel ide illeszti a fájlnév fájlt, amelyről menten ki is ír egy kis statisztikát (sorok száma, karakterszám). Ne próbáljunk 100 kbájtnál nagyobb levelet küldeni. Ha mégis erre szorulnánk, inkább használjuk a {split} parancsot.

Ha kíváncsiak vagyunk, kaptunk-e levelet, csak annyit írjunk: {mail}, és zárjuk le a sort. Ha nincs levelünk, akkor ezt a gép jelzi, míg ha van, egy listát ír ki. Levelekre számokkal hivatkozhatunk, ezek szerepelnek a sorok elején. Ezek után a levél típusa szerepel: új (N), még olvasatlan (U), stb. Ezek után következik a levél küldője és a tárgya. Ha el akarunk olvasni egy levelet, annak a számát kell begépelni, majd lezárni a sort. Ha ki akarjuk menteni a levelet, akkor a '{s} levél_száma fájlnév', ha törölni akarjuk '{d} levél_száma'. Az utóbbi két esetben a levél_száma nemcsak egy szám lehet, hanem például 2-5, ami a 2, 3, 4, 5 leveleket jelenti. Kilépés: a {q} paranccsal. Az elolvasott, de ki nem mentett leveleket a rendszer az {mbox} fájlba menti ki.

A VMS operációs rendszeren is az első dolog a bejelentkezés. Ezután ad-

juk ki a {mail} parancsot. Ezek után a {MAIL} prompt jelzi, hogy a levelezőrendszeren belül vagyunk. Ha már egy fájlban van az elküldendő levél, akkor a '{mail} fájlnév' parancsot adjuk ki; ha csak most akarjuk megírni, elég a {mail} is. A gép most megkérdi a címzettet. Ha ugyanezen gép másik felhasználójának akarunk levelet írni, elég csak a login nevet megadni. Ha távolabbi gépre írunk levelet, kérdezzük meg a rendszergazdától a pontos címzést. Nálunk ez így néz ki: 'MX%"internet_cím"'. Van olyan hely, ahol pedig 'IN%"internet_cím"', ahol az Internet-cím a szokásos loginnév@gép_név alakú. Ezek után a Subjectet kéri a gép, majd kezdhetjük írni a levelet. Ha valamiért nem akarjuk elküldeni a levelet, Ctrl+C-vel félbeszakíthatjuk a levélírást. Ha készen vagyunk, a levelet Ctrl+Z-vel küldhetjük el.

Elindítva a levelezőprogramot, a gép kiírja, hogy hány (új) levelünk van. Ezekről listát a {directory} paranccsal kérhetünk. A levelekre itt is egy számmal hivatkozhatunk. Elolvasásához a levél számát kell megadni. Az aktuális levelet a {delete} paranccsal törölhetjük, és az '{extract} fájlnév' paranccsal menthetjük ki fájlba.

Mindkét esetben, amikor a címzettet kellett megadni, megadhatunk több címet is, szóközzel elválasztva. Ekkor a levelet minden címzett megkapja. Ha nagyon gyakran írunk ugyanarra a címre, felmerül a kérdés, le kell-e írni mindig a teljes címet? Szerencsére nem. Majdnem minden rendszerben létezik az {alias} fogalma, ahol is rövidítéseket adhatunk meg. Például a Unixon az 'alias laci aszalos@llab.arts.klte.hu' parancsot betéve a {.mailrc} fájlba, a {mail laci} ugyanazt fogja jelenteni, mint {mail aszalos@llab.arts.klte.hu}. Aliasok definiálásánál is megadható több cím, így például egy tanár a {mail angolosok} kiadásával minden angol szakos diákjának külön-külön elküldheti ugyanazt a levelet (ha már beállította az aliasokat).

A hagyományos levelekben gyakran előforduló sablon: „Az előző leveledben az kérdezted, hogy ... " Nincs ez másképpen az elektronikus leveleknél sem. Itt azonban könnyebb a helyzetünk, nem kell beírni az idézetet, hiszen majdnem mindegyik levelezőrendszer ismeri a következő lehetőséget: a megválaszolásra váró levél mindegyik sora elé beszúr egy jelet (általában a >-t), s ezután a felesleges sorokat törölhetjük, a válaszainkat a kérdések közé szúrhatjuk. Erre egy példa található a lemezmellékletben (valasz.lev).

Ha valakinek valami nem sikerül, tanulmányozhatja a kézikönyvet a {man mail}, illetve a {help mail} parancsokkal. De még jobb, ha egy hozzáértőt megkér, segítse át a kezdeti nehézségeken.

Ha valaki önállóan tovább szeretne lépni, küldjön egy levelet a {fileserv@@shsu.edu} címre. A Subject sort hagyja üresen, s a levél csak a következőket tartalmazza: '{SENDME MAASINFO}'. Rövid időn belül válaszként 9 levél érkezik, amelyek ékes angol nyelven egyrészt nagyon sok címet adnak, ahol még többet meg lehet tudni a hálózatokról, valamint itt is találhatunk egy kis olvasnivalót.

Aszalós László

APRÓHIRDETÉSI MEGRENDELŐLAP

Kérem, hogy az Új Alaplap következő számának Mikrobazár rovatában közöljék az alábbi szövegű apróhirdetést:

....

....

....

....

....

(Maximális terjedelem: 300 betűhely)

Előfizetés az Új Alaplapra

Az 1994/...... számtól kezdődően előfizetem az Új Alaplap című, havi számítástechnikai folyóiratot

...... példányban ☐ 1 évre ☐ 1/2 évre

(1994-ben a kedvezményes előfizetési díj egy éves időtartamra példányonként 2 820,– forint.)

Az előfizetési díj kiegyenlítéséhez:

☐ **Számlát** kérek (Banki átutalással fizetek)

☐ **Átutalási postautalványt** kérek

Név: ....

(Cég:) ....

Cím: ....

Helység: ....

Irányítószám: ....

Dátum: ....

- - - - - - - - - - - - - - - - - - - -

/aláírás/

MEGRENDELŐLAP

Megrendelem utánvétellel az Alaplap kiadványsorozatokban megjelent alábbi műveket:

ALAPLAP KÖNYVEK

... pld: Jodál Endre: Általános fogalmak (Számítástechnikai alaplexikon I. 3. kiadás) 496,–
... pld: Jodál Endre: Adatkommunikáció és számítógép-hálózatok (Számítástechnikai alaplexikon II.) 356,–
... pld: Buzás Gábor: Ipari számítástechnika (Számítástechnikai alaplexikon III.) 496,–
... pld: Jodál Endre: Mesterséges intelligencia (Számítástechnikai alaplexikon IV.) 496,–
... pld: Kis János: BBS — avagy az elektronikus postaláda (lemezmelléklettel) 656,–
... pld: Jodál Endre: Informatikai alapszókincs 356,–
... pld: Csórián Sándor: Számítógépes kommunikáció 356,–
... pld: Detrik Péter: Az SQL nyelvről 375,–
... pld: Miller László—Tamási Gábor: Macintosh 999,–
... pld: Dárdai Árpád: Mobil távközlési rendszerek 999,–
... pld: Varga Zsigmond: Nyomtatók 999,–

ALAPLAP LEMEZEK

... pld: Norton Guide keretprogram (leírás) 500,–
... pld: PathMinder segédprogram (leírás) 500,–
... pld: CSProlog nyelv (leírás) 1000,–
... pld: LIM EMS 4.0 memóriakezelő (leírás) 1000,–
... pld: Nagy Krisztina: Fractal Generator (program) 1000,–
... pld: Vicsek Mária—Vicsek Tamás: Fraktálnövekedés (program) 1000,–

INFORMÁCIÓKÉRÉS

Kérem, hogy az itt általam **BEKARIKÁZOTT KÓDSZÁMÚ** hirdetésekkel kapcsolatban küldjenek részemre bővebb tájékoztatást.

A0301	A0315	A0329
A0302	A0316	A0330
A0303	A0317	A0331
A0304	A0318	A0332
A0305	A0319	A0333
A0306	A0320	A0334
A0307	A0321	A0335
A0308	A0322	A0336
A0309	A0323	A0337
A0310	A0324	A0338
A0311	A0325	A0339
A0312	A0326	A0340
A0313	A0327	A0341
A0314	A0328	A0342

Beküldhető: 1994. április 30-ig

ÚJ ALAPLAP 1994/3 MÁRCIUS

Feladáskor kérjük bérmentesíteni!

FELADÓ

A) Egyéni érdeklődő:

Név: ....

Cím: ....

Helység: ....

Irányítószám: ....

B) Vállalati érdeklődő:

Cég: ....

Ügyintéző: ....

Cím: ....

Helység: ....

Irányítószám: ....

Telefon/Fax: ....

Bélyeg helye

Új Alaplap
szerkesztősége
Pf. 571
Budapest
1538

Minden PC-hez kell egy jó alaplap!

És egy Új Alaplap!

Belföldön díjmentesen feladható

Új Alaplap
szerkesztősége
Pf. 571
Budapest
1538

FELADÓ:

Név: ....

Cég: ....

Utca, házszám: ....

Helység: ....

Irányítószám: ....

Telefon/Fax: ....

Belföldön díjmentesen feladható

Cédrus Kiadó
Pf. 74
Budapest
1441

Feladáskor kérjük bérmentesíteni!

FELADÓ

Név: ....

Cím: ....

Helység: ....

Irányítószám: ....

Telefon: ....

❑ A hirdetés egyéni és egyedi jellegű, ezért kérem ingyenes megjelentetését. Kijelentem, hogy annak tartalma nem sérti senki szerzői jogát.

❑ A hirdetés kereskedelmi célt szolgál. Mellékelem a soronként (60 karakterenként) 300 forintnak megfelelő összeg átutalásáról az igazoló szelvény másolatát. A címzett: Új Alaplap, 1538 Budapest, Pf. 571, illetve átutalásnál: Agrobank 219-93789/2249-6368

Bélyeg helye

Új Alaplap
szerkesztősége
Pf. 571
Budapest
1538

A LEMEZMELLÉKLET TARTALMA:

- IDE merevlemezek azonosítója DOS és Windows alatt — IDESTUB.EXE, MYIDE#.EXE (Kiss Ferenc)
- C++-példaprogramok — CPLUS#.EXE (Nagy Sándor)
- Kódkonverzió dBase-ben: → 852 → CWI → Ventura — DBKK#.EXE, D~BKK.TXT
- Dinamikus-programozási magyarázatok és programrészletek — D~INPR.TXT (Vargha Dénes)
- Mintalevelek E-mailhez — SUN.LEV, VALASZ.LEV, VAX.LEV (Aszalós Zoltán)
- A véletlen Ctrl+Alt+Del ellen — CSAD02.COM, R~EADME.TXT
- Egy változatos képernyőkímélő — BLAZE#.EXE
- Nyitva van a Csillagkapu! Logikai játék — STARGATE.EXE, S~T.TXT



Hírháló — először, de nem utoljára

A januári Unix-nyitány után ismét új rovat jelentkezik az Új Alaplapban. A hálózatokkal kapcsolatos hírek ezentúl csokorba kötve, rendszeresen jelentkeznek — Kovács Attila jóvoltából.

Internet telefonvonalon?

Csipkerózsika-álmából éledt fel a közelmúltban a felsőoktatási hálózatos alkalmazások mind erősebb elterjedését célzó Huninet Egyesület, amely most többek között ELKöb néven, a meglévő elektronikus levelezési kör bővítésére hozott létre projektet. Az ELKöb keretében olyan programrendszer valósul meg, amely már egyetlen telefonvonal megléte esetén is lehetővé teszi az E-mail szolgáltatás igénybevételét a világ jelenlegi legnagyobb számítógép-hálózatán, az Interneten. A projektbe bekapcsolódó közép- és felsőfokú oktatási intézmények az Internet levelezési átjáróinak segítségével egyéb hálózatokat (pl. CompuServe, Bitnet) is elérhetnek. A legegyszerűbb esetben, a P-mail program használata révén, egyetlen PC bonyolítja a levélforgalmat.

Bonyolultabb, ha a telefonvonalra kapcsolt PC csak levelezési átjáró, amely a beérkezés után egy Novell szerverre továbbítja az üzeneteket. A projekt keretében a modemek ingyenes vagy kedvezményes beszerzését tervezik, s az ELKöb-fejlesztés következő lépcsői: egységes P-mail felület megteremtése Unix-, VAX/VMS-felhasználóknak, továbbá végpontok illesztése X.25-re. További információt Tárkányi Csongor (251-5228) vagy Nagy Gábor (166-4011/2637-ös mellék), a projekt technikai tanácsadója ad.

Aktív elemek a Rolitrontól

Számos újdonság jelent meg a Cabletron- és Wellfleet-disztribútor Rolitron Informatika Kft. azon partnereinél, amelyek aktív hálózati elemeket kínálnak a piacon. Ilyen a Cabletron cég Ethernet elemei között a modulárisan 130 pontig bővíthető HubSTACK eszközcsalád.

Már több helyen működik az országban a MultiMedia Access Centerbe építhető négyportos bridge (EMME), amelynek megjelent továbbfejlesztett változata, az EMME-E kártya. A Cabletron Token Ringhez kapcsolódó termékein belül megjelent az eddig multimódusú kártyák (max. 1-2 km) monomódusú verziója, 10-15 km hatótávolsággal. Piacra került a 40 km hatósugarú monomódusú FDDI/Ethernet bridge kártya is. Az árlistában már megjelent a TP-PMD szabványnak megfelelő, UTP porttal rendelkező koncentrátorkártya, amely az 5-ös kategóriájú UTP kábelen 100 méterig biztosít átvitelt.

Personal NetWare és társai

Roadshow-t rendezett a Novell Közép- és Kelet-Európai Központja Budapesten, amelynek középpontjában új termékek bejelentése állt a hazai felhasználók, a három disztribútor és a szintén képviselt 56 feljogosított viszonteladó képviselői számára. A sajtóanyagok nyolc fontos újdonságról adtak hírt. Ezek közül emeljük ki most a NetWare-rel kapcsolatos legjelentősebbeket. Megjelenik a Novell pont-pont(!) új generációs operációs rendszere, a Personal NetWare, amely üzleti tevékenységet folytató kis cégek és munkacsoportok számára ideális megoldás, egyszerűen menedzselhető és skálázható rendszerszoftver. A termék nagy előrelépés a NetWare Lite 1.1 verzióhoz képest, annak frissítéseként is értelmezhető, könnyű integrálást biztosít a NetWare 2-es, 3-as és 4-es verziójú termékekhez (a NetWare Core Protocolokon át), az SNMP agent bármely SNMP konzolhoz (beleértve a NetWare Management Systemet) hálózati információkat nyújt, egyszerűsítve a hálózatadminisztrációt és -menedzsmentet, kompatibilis a DR-DOS 6.0-val, az MS-DOS 3.0 vagy afölötti változataival, illetve a Windows 3.1-gyel. Megjelent a NetWare Client for DOS/Windows új, 1.1-es verziója, amely a frissített Virtual Loadable Module (VLM) technológián alapul, és univerzális NetWare-kliensként funkcionál. Kis és nagy Novell-felhasználóknak egyaránt megengedi a NetWare hálózati szolgáltatások elérését, beleértve a NetWare 2-es, 3-as és 4-es kiadásait és a Personal NetWare-t. A Novell azt is bejelentette, hogy azok a felhasználók, akik április 30. előtt regisztráltatják NetWare 4 operációs rendszerüket, július 31-ig minden programfrissítést egy ún. Update CD-n ingyen kaphatnak meg. Egyúttal a NetWare 3.11 változatnál a 3.12-re való upgrade-árak 50%-os csökkentését is bejelentette a Novell. Hírek szerint idén tavasszal megjelenik a NetWare on Alpha (a DEC mikroprocesszorán futó verzió) és a NetWare Navigator 3.0.

Racal, Retix és Accton

A Veszprémi Egyetemen a teljes intézményrendszert átfogó hálózati megvalósításra (9 különálló épület között 40 szegmens, 6-700 — a későbbiekben 2500 — csomópont) kiírt pályázatot a Walton Networking Kft. nyerte meg. A hálózat kialakítása optikai kábelezéssel történt. A 100 Mbit/s-os üvegszálas gerincre 4 nagyobb csomópontot alakítottak ki TDM eszközökkel (a Racal-Datacom cég PremNet 5000 típusú berendezéseivel), ehhez kapcsolódnak a Retix RX-7000 típusú útvonalválasztói, amelyek nagy sebességű multiprotokoll-routerek, és szükség szerint Accton típusú multiport-repeaterek. A hálózati fejlesztés bekerülési összege jelenlegi állapotában (Racal SNMP hálózatmenedzsment szoftverrel együtt) 17 millió forint. A Retix routerek a TCP/IP, SPX/IPX és DECnet protokollokat egyaránt támogatják. A február közepén átadott hálózati rendszer minden eleme (építőelemek, intelligens eszközök, munkaállomások) egy központi helyről felügyelhető. A Racal, Retix, Accton termékek hivatalos magyarországi disztribútora a Walton.

Új 3Com(binációk)

Új routercsaládot alkotnak a 3Com cégnél a NETBuilder Remote Office termékek. Az önállóan és a NETBuilder II., illetve NETBuilder Remote routerekkel egyaránt használható hálózati útvonalirányítók tipikusan a kisebb távoli irodák közötti, Ethernet vagy Token Ring technológián alapuló adatforgalom eszközei. Újdonság a LinkBuilder „stack-elhető” hub-családjában az igen előnyös árfekvésű 24 portos 10BaseT hub. A felhasználók vezérelhetik az új hubot a 3Com Transcend LinkBuilder FMS Manager for Windows nevű alkalmazási programjával. Időközben a 3Com felvásárolta a szintén amerikai Synernetics-et, az Ethernet és FDDI kapcsolóeszközök és technológiák vezető cégét. Ezzel a 3Comnak az a célja, hogy az Ethernet kapcsolóeszközök vezető gyártójává váljon.

INFORMÁCIÓKÉRÉS: A0330 ▲

Mesterséges mesterek — II.

A „művezető" megválasztása

A robottechnika interdiszciplináris tudományterület. Ezúttal a robotok irányításának problémakörét, az egyes módszereket foglaljuk össze.

A mai robotok ősének történelmi távlatokban tekintve voltaképp az egyszerű emelők, majd később Arkhimédész csigái tekinthetők. Ezeket emberi vagy állati izmok erejével működtették. A fejlődés további fázisát a gőzgépek, belsőégésű motorok, pneumatikus, hidraulikus és elektromos motorok megjelenése, majd ezen eszközök vezérlésének tökéletesedése, ezáltal azok nagyobb termelőegységbe (rugalmas gyártórendszerekbe) való beépítése jelentette. Emiatt a robottechnika mai fogalmaink szerint a következő szakterületek integrált alkalmazását jelenti:

— **Gépészet,** amely magában foglalja a különböző szerkezeti elemek (karok, mechanikai áttételek, megfogók, szerszámcserélők, motorok és egyéb hajtások, fékek) tervezését, alkalmazását.

— **Informatika vagy információtechnológia** célja az egyes robotok vezérlésének megoldása a nagyobb termelőegységen — a rugalmas gyártócellán — belül. E terület durván a robot és a cella közti információ kódolására, a különböző egységek közti kommunikáció hatékony szabályainak megállapítására terjed ki.

— **Szenzortechnológia,** azaz különböző külső és belső érzékelők tervezése és használata. Belső érzékelők például: az egyes karszakaszok egymáshoz képesti helyzetének valós idejű mérésére szolgáló lineáris vagy forgótárcsás, relatív és/vagy abszolút elmozdulást érzékelő optikai rácsok vagy elektromágneses működési elvű kódadók, egyszerű mechanikus végálláskapcsolók, elektromágneses vagy optikai közelségérzékelők, a kontakt erők és forgatónyomatékok mérésére szolgáló nyúlásbélyeges vagy piezoelektromos elvű mérőeszközök, amelyek a robotkarok belső állapotáról szolgálnak információval. Külső érzékelők például: optikai vagy ultrahangos távolságérzékelők, ipari látómodulok, erő- és nyomatékmérő eszközök, biztonsági közelségérzékelők, amelyek a robot környezetének állapotát figyelik. E szenzorok jeleinek feldolgozása (zajszűrés, nemlinearitások és torzítások figyelembevétele stb.) szintén az informatika tárgykörébe tartozik.

— **Irányítástechnika,** azaz a robot+munkadarab vagy robot+technológiai szerszám mint fizikai rendszer modellezése, s különböző módszerek, vezérlési és szabályozási stratégiák kidolgozása annak érdekében, hogy e komplex és bonyolult fizikai rendszert valamiképp „rábírhassuk" a gyártási folyamatban szükséges előírt mozgások és technológiai műveletek elvégzésére.

Az MI speciális alkalmazási területe

A mesterséges vagy gépi intelligencia, azaz e rovat közelebbi tárgya a fenti szakterületek közül mindegyikben megjelenik, bár azokban annak inkább különböző oldalai, vonatkozásai dominálnak, domborodnak ki. Az egyik érdekes szakterülettel, a gépi kép feldolgozásával vagy a látás automatizálásával több cikk részletesen foglalkozott. Az egyik speciális megoldási mód, a mesterséges neurális hálózatok működése és alkalmazása szintén kellő nyomatékot kapott a rovatban. Tekintettel arra, hogy a robottechnika sokkal szélesebb területet ölel fel annál, mint amekkorát egy valamilyen szakterületre specializálódott szakember egyáltalán kellő mélységben át tud tekinteni, jelen cikkben csak az előbbi felsorolás végére hagyott irányítástechnikának — mint az MI-alkalmazások jelentős terének — általam valamennyire ismert részterületeihez tartozó problémák, illetve azokból származó példák ismertetésére szorítkozunk. (Ez természetesen nem jelenti azt, hogy ez a részterület lenne a mai robottechnikai kutatások legfontosabb ágazata, a gépi intelligencia fontosságának illusztrálására viszont ezen belül is bőségesen találunk példát.)

A robotirányítási feladat alapelemei

A robotok irányítása mint műszaki probléma az alábbi elemeket foglalja magában.

a) Technológiai feladat

A robot által végrehajtandó technológiai feladat a robot megfogójában („kezében") lévő munkadarab vagy szerszám valamilyen módon előírt pályán mozgatásából, a szerszám ki-be kapcsolásából, illetve bizonyos pontokban meghatározott kontakt erők és forgatónyomatékok kifejtéséből áll.

b) Kinematikai pályatervezés

Rendelkezésünkre, illetve a robot vezérlőegységének rendelkezésére áll valamilyen — a gyakorlatban majdnem egzaktnak tekinthető pontossággal ismert — tudás a robot karjának kinematikai szerkezetéről. Ez az ismeret alapvetően a robot „belügye" abban az értelemben, hogy a végrehajtandó feladat általában a felhasználó kényelmes tájékozódását lehetővé tevő ún. műhely-koordinátarendszerben van megadva (pl. egy kifúrandó furat helye a munkadarab tervrajzán, vagy ennek CAD-rendszerrel létrehozott ekvivalensén). A robot vezérlése viszont „kénytelen" a robot általános koordinátáiban, azaz az egyes karszakaszok belső érzékelőkkel mért, egymáshoz viszonyított elfordulásainak vagy lineáris transzlációinak adatrendszerében „gondolkodni". Ha az x szimbólummal a műhelyrendszerben megadott adatokat mint többkomponensű, $x=(x_1,...,x_m)$ formátumú tömböt, a $q=(q_1,..,q_n)$ szimbólummal pedig a robot általános koordinátáit — ennek elemei szög-, ill. elmozdulásértékek — jelöljük, ez az ismeret egy viszonylag áttekinthető szerkezetű nemlineáris x(q) egyértelmű függvénnyel reprezentálható, amely általában nem invertálható egyértelműen, ezenfelül pedig e nem egyértelmű inverz csak speciális szerkezetű karok esetében fejezhető ki a néhány alapfüggvény segítségével véges, zárt alakban. Mint láttuk, a gyakorlat a nehezebb, q(x) ún. inverz kinematikai probléma megoldását kívánja meg. Amennyiben erre már van valamilyen

hatékony módszerünk, meg tudjuk oldani a kinematikai pályatervezés feladatát.

c) Dinamikai feladat

A kinematikai pályatervezés elvégzése korántsem jelenti a vezérlési probléma megoldását. A bonyolultabb technológiai feladatokban meghatározott erők és forgatónyomatékok kifejtése is elő van írva; ezenfelül az ugyanazon munkadaraboknak vagy szerszámoknak ugyanazon a pályagörbén mozgatása — az előírt sebességtől függően, a mechanika törvényei szerint — ugyanazon pontokban különböző erő-, illetve nyomatékkifejtést tesz szükségessé. Hasonló a helyzet a robotkar egyes darabjainak a mozgatásával, amelyeknek szintén jelentős saját tömegük, tehetetlenségük van. Ezzel eljutottunk a robotirányítás dinamikai aspektusaihoz.

A robot kinematikai szerkezete által meghatározott módon a robot vezérlése által beállítható értékek a $q=(q_1,..,q_n)$ általános koordináták tengelyeit a mozgatómotorok által kifejtett $Q=(Q_1,..,Q_n)$ általános erők. Ezek komponensei fizikai értelemben erők és/vagy vegyesen forgatónyomaték dimenziójú mennyiségek. Ha q(t)-vel jelöljük az inverz kinematikai feladat megoldásával kapott mozgásgörbét, akkor azt írhatjuk, hogy e Q általános erőkomponensek egy meglehetősen bonyolult, nemlineáris

$$Q=Q(F,q,dq/dt,d^2q/dt^2|P)$$

függvénnyel jellemezhetők. Ebben F a technológiai műveletben előírt erőkre és nyomatékokra utal, P pedig egy paramétertömb, amely az egyes karszakaszok és az éppen mozgatott szerszám vagy munkadarab általában nem pontosan ismert és (a munkadarab felvételekor, illetve elengedésekor) időben ugrásszerűen változó inerciaadatait reprezentálja.

d) Valós idejű ellenőrzés, visszacsatolás

Tekintettel kell lennünk még arra, hogy a $Q=Q(F,q,dq/dt,d^2q/dt^2|P)$ függvényben P általában nem pontosan ismert. Ezért, ha a robot irányítása egyáltalán foglalkozik annak modellezésével, adott Q beavatkozó jel nem pontosan a feltételezett d^2q/dt^2 gyorsulásértékeket, következésképpen nem a pontosan kívánt q(t) mozgásfüggvényt eredményezné. Azért, hogy ezek a hibák ne gyarapodhassanak vég nélkül a rendszer mozgásában, általában szükséges az elvárt és a megvalósuló hatás egymástól való eltérésének figyelése, és ennek megfelelően korrekciók, azaz visszacsatolás alkalmazása.

Az egyszerűbb irányítási eljárások esetén pusztán egy ilyen visszacsatolás megakadályozhatja a hibák vég nélküli növekedését, és azokat képes véges korlátok között tartani. Konstruálhatók olyan „drasztikus" irányítási stratégiák, amelyek bizonyos értelemben érzéketlenek az alkalmazott modell pontatlanságaival szemben. Ezek az ún. „robusztus" irányítási stratégiák.

Robusztusan a robotot...

Az irányítás robusztus bizonyos modellezési hibákkal vagy külső zavarokkal szemben, ha annak minősége nem változik meg lényegesen e hibák vagy zavarok miatt. A robusztusság tágabb értelemben valamilyen kvalitatív tulajdonság (pl. konvergencia) megmaradását jelenti a paraméterek folytonos változása esetén is, míg e tulajdonság kvantitatív jellemzői (pl. a konvergencia sebessége) folytonosan változhatnak.

A beavatkozó jelre adott tényleges és az alkalmazott modell alapján várt rendszerválasz eltérésének analízise módot adhat e modell pontosítására vagy lényegesebb módosítására az ún. adaptív irányítások esetében. Az irányítás adaptív, ha alkalmazkodik a vezérelt rendszer viselkedésének megváltozásához — annak működése minőségi romlásának megakadályozásával, korlátok közt való tartásával vagy csökkentésével —, függetlenül attól, hogy e viselkedést a rendszer természetes belső tulajdonságainak (paramétereinek) változása és/vagy a működtetés környezetének időben nem állandó hatása váltotta ki.

Az irányítás minősége sokféleképp előírható fogalom. A pontos matematikai megfogalmazás itt nem mindig előnyös, mert miatta a szabályozás működésének értékelése és maga a vezérlés is fölöslegesen komplikálttá, nehézkessé, működése időigényessé válhat. A gyakorlatban ezért az irányítás minősége sokszor lehet egy kvalitatív vagy „fuzzy" (azaz „életlen") jellegű fogalom. (E cikksorozatban később a fuzzy halmazokat pontosan definiáljuk.)

Érdemes megjegyezni, hogy a robusztusság lehet az adaptivitás vagy egyéb, konstrukciós jellegű elvek betartásának következménye.

Fizikai modellalkotási problémaként

Mind az élő állatok, mind az ember, mind a gépek intelligenciájának közös vonása, hogy ezek az entitások valamennyien rendelkeznek az őket körülvevő világról és önmagukról alkotott valamilyen képpel, amelynek alapján az őket kívülről érő ingerekre adott válaszaikat kialakítják. Ezt a világképet általában csak az ember esetében szokás tudatosnak nevezni. Cikkünkben akár szigorú matematikai modellalkotáson, akár egyéb tudatos, akár nem tudatos „világkép"-ről van szó, azt modell névvel illetjük. A modellalkotást pedig — mint a valóság megközelítő leírását — fizikai aspektusból tekintjük.

Az irányított rendszerre vonatkozó modell egy gyakorlati esetben nem lehet matematikailag nagyon bonyolult, hiszen ebben az esetben az a valós idejű alkalmazásra alkalmatlan lenne. (Túl sok és bonyolult számítást nem lehet nagyon rövid idő alatt elvégezni.) Ezért a modellalkotás folyamán a robotot általában mint géprendszert — azaz mint több, önálló funkcióra alkalmas egységből álló, koordinált működésre és előírható feladatok ellátására képes irányított gépi konfigurációt — értelmezzük. E megközelítés kiváló alapul szolgál annak a matematikai modellalkotási módnak, amely szerint: egy áttekinthetetlenül bonyolult, összefüggő rendszert többé-kevésbé áttekinthető és a többi résztől függetlenül modellezett részrendszerekre bontunk — s e részek egymásra hatását kontakt erők, forgatónyomatékok, impulzus-, energia-, anyag-, információ- stb. folyamok (mint kimeneti, ill. bemeneti mennyiségek) bevezetésével kívánjuk leírni.

Maga az irányítás

A legáltalánosabb szinten az irányítás a következőkből áll: a géprendszer egészére előírt komplex feladatnak az egyes részrendszerekre kiróható feladatokra bontása; e részfeladatok közlése a hozzájuk tartozó részrendszerekkel; a részfeladatok végrehajtásának ellenőrzése.

A kirótt feladat végrehajtásának ellenőrzése esetenként elmaradhat, vagy lehet rendkívül primitív — pl. ha csak annak vizsgálatára szorítkozik, hogy a kívánt végeredmény bekövetkezett-e. Az ilyen jellegű irányításokat vezérlésnek nevezzük. Ha a végrehajtás ellenőrzése igen szofisztikált, a folyamat részleteinek időben folytonos figyeléséből, s ennek megfelelő beavatkozó korrekciók alkalmazásából áll, az irányítást szabályozásnak nevezzük. (Mindhárom szó angol fordítása a sokkal általánosabb „control".)

E koncepciónak megfelelően a robotok vezérlése általában egy központi

vezérlőegységre, és az egyes q_i komponenseket közvetlenül meghajtó motorok lokális hajtásszabályozásaira bontható fel. Ebben a konstrukcióban az egyes tengelyek mozgásának (akár dinamikai, akár kinematikai) koordinálása a központi egység feladata, míg az egyes lokális, ún. tengelyszabályozások a bonyolult nemlineáris csatolásokkal már „nem foglalkoznak", hanem az ezek következtében fennálló hatásokat mint egyszerű hibát csak visszacsatolják az adott tengely független szabályozóegységének bemenetére.

E hierarchikus struktúrában a központi egység (digitális realizációban legalább 20 ms vagy ennél rövidebb) időközönként vezérlőjelet állít elő az egyes tengelyek hajtásszabályozásai számára. Ez utóbbiak vagy analóg működésűek, vagy digitális szervezés esetén a központi egység időfelbontásánál legalább egy nagyságrenddel finomabb időfelbontással megkísérlik a kiadott feladat végrehajtását. A központi vezérlőegység egy időciklusa alatt felgyűlt hibát a rendszer visszaolvassa a robot belső érzékelőiről, s ezek csatolásokon keresztüli korrigálása a következő ciklus alatt kezdődik meg.

Nem szóltunk eddig arról, hogy a központi vezérlőegység jelei fizikailag milyen mennyiségnek felelnek meg. Ennek függvényében lényegében ugyanaz a hierarchikus vezérlési struktúra a rendszerről alkotott többféle modell szerinti irányítás megvalósítására lehet alkalmas. Feltéve, hogy az alkalmazott hardver számítási, illetve adatátviteli kapacitása elég nagy, az aktuális működési mód a vezérlésre használt szoftver (a gépi intelligencia egyik fontos hordozó eleme) függvénye.

Maguk a robotkarok mechanikai testek, amelyek mozgásának gyorsulását (a d^2q/dt^2, idő szerinti második deriváltakat) lehet beállítani rájuk ható erőkkel vagy forgatónyomatékokkal. Fizikai értelemben a fenti hierarchikus struktúra akkor működne „egzaktul", ha a központi vezérlés erő- és nyomatékjeleket számítana ki, s az egyes hajtásszabályozások az ilyen értelmű utasítások végrehajtására lennének képesek. Ez felel meg az ún. „kiszámított erő vagy nyomaték szerinti szabályozásnak", amelyet a szakirodalomban Computed Torque Control néven tartanak számon. Mivel azonban a nyomatékok számítása igen bonyolult, és sok számolási munkát igényelne (ráadásul csak részlegesen és közelítőleg ismert paraméterek mellett), az iparban kezdetben ennél sokkal kevésbé adekvát megoldások terjedtek el.

Vezérlések/szabályozások

A kezdeti alkalmazásoknál beérték az ún. pontvezérléssel (Point To Point vagy PTP). Ennek lényege, hogy a robot számára csak a karok kívánt kezdeti és végpozíciója van megadva. A robot feladata kimerült abban, hogy a kezdetiből a végpontba „a vezérlés tetszése szerint" kialakított pályán mozogva érjen el, és ott álljon meg. E megoldásban a hajtásszabályozások bármiféle egymás közti koordináció nélkül, maguk döntötték el, hogy az egyes tengelyeket hogyan mozgatják a kezdeti pontnak megfelelő értékről a végpontnak megfelelő értékre.

A felhasználó számára a robot mozgásának pályagörbéje ellenőrizhetetlen volt a két pont között. A mozgás finomítására csak a hosszabb pálya rövidebb szakaszokra való felosztásával volt mód.

A gyakorlati alkalmazások széles köréhez ez a módszer nem bizonyult eléggé kifinomultnak, ezért hozták létre az ún. pályavezérlés (Continuous Path vagy CP) módszerét. A központi vezérlőegység nemcsak a kívánt végpont koordinátáit adja meg a hajtásszabályozásoknak, hanem magára vállalja a kívánt pályagörbe egyenletes időközönkénti felosztását, és az ennek megfelelő pontokban a kívánt $q(t_i)$ koordinátaállások kiszámolását. A hajtásszabályozások feladata ezen értékek elérése (ún. koordinátaszabályozás — Coordinate Control).

A módszer tovább finomítható, ha a mozgást a (q,dq/dt) fázistérben próbáljuk reprezentálni. Ez szigorúbb kötelem, mint amit a koordinátaszabályozásnál írunk elő, hiszen nemcsak azt rögzítjük, hogy az adott t_i időpontokban a robotnak a $q(t_i)$ pontok közelében kell tartózkodnia, hanem azt is, hogy e pont környékén ráadásul $dq(t_i)/dt$ sebességgel kell haladnia.

Látható, hogy a központi egység és a lokális hajtásszabályozó egységek közti adatforgalom ekkor megnő, hiszen nemcsak koordináta-, hanem sebességértékekkel is foglalkozni kell (ún. sebességszabályozás — Resolved Motion Rate Control). Világos, hogy mechanikai értelemben egyik módszer sem adekvát, hiszen bár a hajtások motorjai ekkor is forgatónyomatékokat fognak kifejteni a robotkar egyes részeire, ezek a nyomatékértékek nem a mechanika törvényei szerint vannak meghatározva.

A klasszikus mechanikai értelemben vett korrektség felé vezető lépés az ún. gyorsulásszabályozás (Resolved Acceleration Control), amelynél a központi vezérlés egy $d^2q/dt^2=f(q,dq/dt)$ típusú függvény kiszámolására törekszik — kellően sima pályagörbe létesítése céljából. Amennyiben e függvény számolása a robot+munkadarab rendszer valamilyen közelítő dinamikai modelljén alapul, elérkeztünk a nyomatékszabályozáshoz.

Figyelembe kell még vennünk, hogy a fenti irányítási lehetőségek „tisztán" csak akkor alkalmazhatók, ha a robotkar szabadon mozgathat egy munkadarabot. A technológiai folyamatoknál azonban nem ez a helyzet, mivel a szerszám mechanikailag érintkezik a munkadarabbal. Ez esetben például a munkadarab felületének adott pontbeli érintősíkjával párhuzamosan előírhatunk elmozduláskomponenseket, e sík normálisa irányában viszont nem, hiszen a szerszám általában nem merülhet bele a munkadarabba. A normális irányában viszont előírható az érintkezéskor ható erő. Miután ez a módszer az erő- és a koordinátaszabályozás elemeit ötvözi egymással, neve hibrid szabályozás.

Tar József

OLTP-modell — tranzakciómonitorral

Impozáns teljesítménynövekedés

Túllépve az elszigetelt PC-s applikációk, majd a lokális hálózatok világán, a feladatok komplexitása rohamosan nő. Komplexebb eszközöket kell használnunk megoldásukra. Közülük most a Magyarországon még kevéssé elterjedt tranzakciómonitorokra hívjuk fel a unixosok figyelmét. Eddig a számítástechnikai szaksajtó keveset foglalkozott ezzel a témával. Pótolandó a mulasztást közreadjuk szerzőnknek az AT&T-vel folytatott levelezéséből származó autentikus információit.

A kilencvenes évek számítástechnikájának legdinamikusabban bővülő része az Online Transaction Processing (OLTP). Ma már igazi húzóágazatnak tekinthető: növekedésének üteme kétszerese a teljes számítógépes ipar növekedésének.

Az OLTP-alkalmazás új fogalom a számítástechnikában, egy olyan applikáció, amely nagy adattömegen biztosít online elérést és feldolgozást. Az OLTP körébe tartozik például egy áruházban a pénztárak terminálhálózata, és OLTP a nagy légitársaságok helyfoglalási rendszere is.

A két példának több közös pontja is van: egyrészt a feldolgozás online módja; másrészt az alkalmazásoknál kiemelkedően fontos az adatok integritása; harmadrészt mindkét esetben minimalizálni kell az esetleges rendszerleállások idejét.

Eltérő számítástechnikai kultúrák

Hazánk és a fejlett számítástechnikával rendelkező országok közel azonos jellegű OLTP-feladatokkal állnak szemben, csak teljesen más környezetben. Nyugaton ma a „downsizing" és a „rightsizing" a kulcsszó: a költséges üzemeltetésű mainframe gépek lecserélése olcsóbb, gyártófüggetlen kis- és középgépekre — Unix operációs rendszerrel.

Nálunk, ha egy alkalmazásnak egyáltalán létezett is számítástechnikai előzménye, az legtöbbször PC-s lokális hálózatot jelentett, amelyet mind a feldolgozás kapacitásában, mind funkcionalitásában kinőttünk.

Manapság növeljük az erőforrásokat, és nemegyszer az alkalmazás geográfiai méreteit is. Nyugaton a kliens/szerver architektúra, a relációs adatbáziskezelők (RDBMS) és a tranzakciómonitorok jelentik a járatos eszközöket. Nálunk a kliens/szerver felépítés már elfogadott, a relációs adatbáziskezelők majd mindegyike jelen van a hazai piacon, a tranzakciómonitorokról viszont alig hallottunk valamit. Pedig ezek a termékek nem ma jelentek meg először. Az AT&T de facto szabványnak számító TUXEDO rendszere például már a 80-as évek elején létrejött.

Egy tipikus mai feladat

Egy hagyományos, földrajzilag egy helyszínen telepített konfiguráció esetén az adatbáziskezelő kliens komponense kommunikál a felhasználóval, és továbbítja az igényeket a szerver felé. A szerver gondoskodik a tranzakciók végrehajtásáról, optimalizálásáról és az adatok integritásáról. A közbülső kommunikációt nagy sebességű, megbízható lokális hálózat biztosítja.

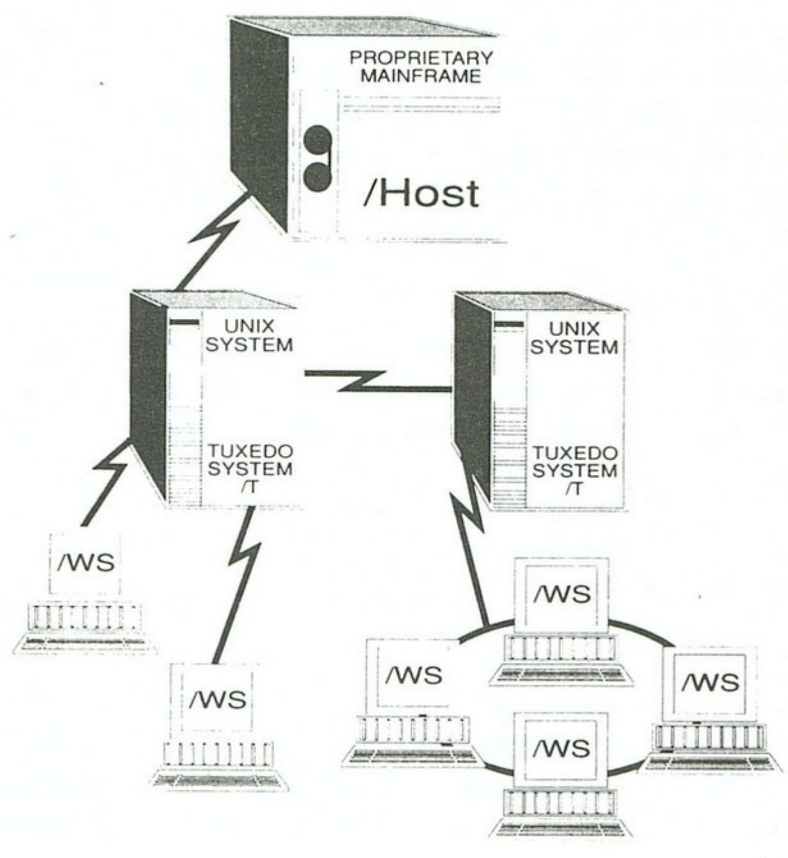
A tipikus mai feladathoz azonban országos méretű, esetleg országhatárokon is túlnyúló topológia tartozik. Jellemzően már meglévő, de egymástól eltérő RDBMS-re épülő adatbázisok integrációját követeli, a rendelkezésre álló hardver- és operációsrendszer-platformok pedig ritkán homogének. A földrajzilag távol eső csomópontokat összekötő kapcsolatokban az adatátvitel kapacitása kritikus. A megvalósítandó topológia ritkán egyszerűsíthető kliens/szerver kapcsolatra, sokkal gyakoribb a végpont-alközpont-központ jellegű, három- vagy többszintes hierarchia. Végül: viszonylag egyszerűen kellene üzemeltetni az egész rendszert.

Ha ezeket a problémákat hagyományos eszközökkel akarjuk megoldani, akkor megnő az adhoc jellegű programozás aránya, és csökken a megbízhatóság is. Tranzakciómonitor alkalmazásával azonban leegyszerűsíthető a feladat, csökkenthető a fejlesztési idő (költség), és nem utolsósorban jóval robusztusabb rendszer hozható létre.

A modell és szintjei

Ezek után vizsgáljuk meg, mi az a modell, amelyet a tranzakciómonitorok segítségével alkalmazhatunk kurrens OLTP-feladatok megoldására.

A konfiguráció legalsó szintjén a felhasználóval kommunikáló eszközök — RISC- vagy PC-alapú munkaállo-



mások — helyezkednek el, operációs rendszerük Unix, Windows, NT, Macintosh vagy OS/2. Ezekre az eszközökre hárul a felhasználói igények előfeldolgozása, és ezek szolgáltatják az eredményt a felhasználónak. Grafikus felülettel dolgoznak, és a jövőben egyre inkább használják a multimédia lehetőségeit is. Mindinkább elterjednek az adatbázis-független front-end eszközök (Gupta, Magic, PowerBuilder, Supernova), amelyeket eleve az ilyen felhasználói csatoló szerepre fejlesztettek ki.

A középső — akár több — szintet a Unix-alapú középkategóriás gépek jelentik, amelyek a munkaállomásokkal lokális hálózaton kommunikálnak, egymáshoz és a legfelső szinthez pedig WAN-okon (X.25 protokollal) kapcsolódnak. Itt futnak a relációs adatbáziskezelők hatékony back-endjei (amelyek más-más csomópontokban eltérő termékek lehetnek), kielégítve a felhasználók többségének feldolgozási igényeit. Azokat az igényeket, amelyeket helyben nem lehet feldolgozni, ez a szint továbbítja valamelyik szomszédjának vagy a legfelső szintnek.

A hierarchia tetején helyezkedik el a nagy kapacitású Unix szerver (Sequent, Pyramid) vagy a „továbbszolgálós" mainframe. Ez tárolja a globális adatokat, és feldolgozza a közbülső szintekről érkező igényeket. Ezen a szinten történik a menedzsmentszintű riportok összeállítása is.

Az alsó, közbülső vagy legfelső szinten telepített tranzakciómonitor jelentősen kibővíti a hagyományos kliens-szerver modell lehetőségeit mind a fejlesztők, mind a felhasználó számára.

Az intelligens szervező

A fejlesztők egy magas szintű API, az Application Transaction Management Interface (ATMI) használatával valósítják meg a kliens/szerver kommunikációt szinkron vagy aszinkron úton, operációsrendszer- és adatbázis-független módon. Az ATMI-vel definiálhatóak atomi tranzakciók, amelyek segítségével a különböző földrajzi helyeken és adatbáziskezelőkön telepített adatok egyetlen logikai adatbázist alkotnak, amelynek integritására a tranzakciómonitor felügyel. Koordinálja az elosztott tranzakciókat (a hálózat különböző pontjain aktuális update-eket), felügyeli a kétfázisú commit protokollt, és garantálja, hogy a commit vagy rollback az egész rendszerben egységesen történjék.

Mivel a tranzakciómonitor „érti" az elosztott adatbázis struktúráját, futás alatt képes a hálózati forgalom intelligens szervezésére, például bizonyos adatokat csak a megfelelő szervereknek „routol". Egyes hálózati kapcsolatok kiesése vagy túlterheltsége esetén alternatív útvonalak aktivizálódnak. Lehetséges a kliensigényeknek az alkalmazás logikája által meghatározott prioritás szerinti feldolgozása: bizonyos típusú tranzakciók — a kritikus tevékenységek — magasabb prioritással hajtódnak végre, mint mások.

Úgynevezett „típusos pufferek" segítségével jelentősen csökkenthető az erőforrásigényes műveletek száma, növelhető a teljesítmény. A már implementált rendszer felügyeletét adminisztrációs és finombeállító eszközök segítik. A hálózati forgalom adatvédelmét például a Kerberos rendszer alkalmazásával növeli.

A jól beállított tranzakciómonitor magas funkcionalitása ellenére sem okoz plusz terhelést a rendszer számára. Sőt, „intelligenciája" révén számos optimalizációt végez futás alatt, így installálása nemegyszer 200-500%-os teljesítménynövekedést eredményez.

Porkoláb Zoltán

Új vizeken az Albacomp

Közel 4 milliárdos forgalmat bonyolított le az Albacomp. Az elsősorban nagykereskedelmi tevékenységet folytató, főleg hardverorientált székesfehérvári cég december elején megkezdte a közvetlen értékesítést is, kiskereskedelmi áruházat nyitott Budapesten. A jól jövedelmező kereskedelem mellett azonban új területre, az alkalmazások és alkalmazásfejlesztési technológiák területére is kiterjeszti tevékenységét.

A felhasználók teljes körű kiszolgálása érdekében konzultációs csoportot hozott létre, amely foglalkozik az igényfelméréstől a vállalat üzletmenetének átszervezésén át (business process reengineering) a számítástechnikai rendszer komplett kivitelezéséig. A Consulting Group új, korszerű alkalmazásfejlesztési technológiákat honosít meg Magyarországon. Ennek első állomásaként az Albacomp felvállalta a SuperNOVA objektumorientált 4GL fejlesztési környezet disztribúciós feladatait, amelyhez értékhozzáadó viszonteladóként csatlakoztak elsőnek a MemoLuX szoftverfejlesztői is.

Társulás a térinformatikáért — polyGIS

A szerteágazó tevékenységeket és tudományágakat összefogó térinformatika mind felhasználói, mind fejlesztői szempontból igen költséges, mert az információk begyűjtésének, feldolgozásának és karbantartásának eszközei nagyon drágák. Különösen igaz ez olyan projekteknél, ahol például egy nagy területet átölelő, nagy méretarányú térképet kell előállítani, és esetleg csatolni kell hozzá a biztonságos és speciális adatbázisokat tartalmazó információs rendszert.

Éppen ezért a honi térinformatikai alkalmazási területek öt vállalata polyGIS néven térinformatikai társulásba szerveződött a rendszerek hazai elterjesztésére. Tagjai (Carto-Hansa, Kangyal, MÁFI, MemoLuX, Rudas & Karig) a szakma különböző részterületeinek specialistái. Összefogásuk az egyik legerősebb itthoni GIS-vállalkozás, amely lefedi a településfejlesztés és üzemeltetés, mezőgazdasági és ipari GIS-alkalmazások, környezeti hatásvizsgálatok, információszolgáltatás, valamint a rendszerintegrátori és gazdasági tanácsadás feladatait.

Bár a polyGIS — eszközparkja és szakembergárdája miatt — eddigi fejlesztéseit zömmel Intergraph-bázison végezte, ez részükről nem jelent elkötelezettséget, mert egyéb környezetekben is komplex térinformatikai rendszereket hoznak létre — biztosítva a kapcsolatot DOS, Windows, Novell és Unix rendszerekkel.

Számításba vett Számalk

A nagy múltú, magyar tulajdonú, nemrég privatizált Számalk képes a megújulásra, sőt fejlődésre is: 1,8 milliárd forintnyi árbevételt ütemeztek elő, és 1,95 milliárdos forgalmat értek el 1993-ban. További 20%-os fejlődéssel számolnak 1994-ben, dinamikus növekedést azonban csak 1996 végétől várnak a vezetők.

Rendkívül megerősödött a cégen belüli disztribúciós tevékenység, ebből származik a cég árbevételének a fele. A Szoftver Disztribúció több mint 450 milliós forgalmából a Microsoft-termékek forgalma meghaladta a 320 milliót — tehát a cégek nem(csak) lopják, hanem veszik is a szoftvert...

A Hardware (Compaq, Calcomp, Sony, miro) Disztibútor csak most ünnepli egyéves születésnapját; 246 millió forintos árbevétele igen szép teljesítmény. Idén forgalmukat szeretnék 400 millióra feltornászni. A Compaq gépekből értékesítették a legtöbbet, ezen belül a ProLinea a legnépszerűbb a magyar felhasználók körében, a Presario azonban nem vált be, pedig Nyugaton kedvelt otthoni számítógép. Nagy számban adtak el monitorokat és harddiszkeket.

A kereskedelem mellett a hozzáadott értéket is növelni akarja a Számalk, ezért hamarosan berendeznek egy olyan Microsoft-Novell-Compaq Oktatóközpontot, ahol vizsgáztatnak is.

Fontos terület — az összforgalom negyede — a rendszerintegrálás, ezt a tevékenységüket külföldi és belföldi szoftverházak és hardverszállítók bevonásával szeretnék megerősíteni. Rendkívül büszkék arra, hogy — a KFKI-val és az MTI-vel közösen — ők építik a Világkiállítás Informatikai Rendszerét.

A Számalk harmadik, talán legismertebb tevékenysége az oktatás, ez adja árbevételük fennmaradó negyedrészét. A 93-as év nagy sikerének könyvelték el a Gábor Dénes Műszaki Főiskola felvirágoztatását, amely távoktatási koncepciójával egyedülálló a magyar felsőoktatásban.

Egy nullkilométeres cég

A tavaly júliusban alakult VAR Kft magas műszaki színvonalú termékek gyors kiskereskedelme mellett LAN-ok átalánydíjas szervizelési feladatait látja el, és — a Nebulóért mozgalom keretében — bekapcsolódott a vidéki általános és középiskolák számítástechnikai eszközökkel való ellátásába. A legismertebb tevékenységük azonban a KO-REC-TYPE festékszalagok és a KAO-lemezek disztribúciójához kapcsolódik.

Az ISO 9002-es szabványnak megfelelő KAO mágneslemezek minden méretben és írássűrűséggel beszerezhetők. Nyugat-Európában a 8"-os lemezek korszaka már leáldozott, az 5,25"-os lemezek iránti kereslet csökkenőben van, és tartósan nő a 3,5"-os lemezek forgalma. A KAO vezetői szerint Kelet-Európában még 2 évet kell várni a 3,5"-os lemezek áttörésére. A nemrég megjelent 4 Mbájtos lemez azonban „nem jött be".

A KAO lemezsokszorosítást is nagy tételben végez, elsősorban a szoftverkészítő cégeknek. A hazai mezőnyből a Graphisoft szoftverei jelennek meg KAO-lemezeken. (De ne feledkezzünk meg az Új Alaplap és a PC World lemezmellékleteéről sem!). Világviszonylatban a KAO a Microsoft legnagyobb beszállítója és szerződött sokszorosítója. A lemezek piacán a KAO világméretekben vezető helyet foglal el, itthon azonban még mindig a 3M lemezekből adnak el többet.

A Montana felnyúlt az égbe...

A közép- és kelet-európai távközlési tevékenységben érdekelt amerikai SFMT és a magyar számítástechnikai és távközlési magáncég, a Montana Holding vegyesvállalatot hozott létre SFMT-Montana Telecom néven. A cég — amely a Montana Telecom tevékenységét viszi tovább — az üzleti élet számára szükséges telekommunikációt szeretné meghonosítani, nemcsak Magyarországon, hanem Kelet-Európa más országaiban is. (A Montana Telecom titkosításból álló profilját a Montana Kriptológia Kft veszi át.)

Az SFMT-nek komoly tapasztalatai vannak a Kelet-Nyugat közötti kommunikációt illetően. (Többek között ők valósították meg az első E-mail kapcsolatot az USA és a KGST-országok között.) 10 millió dollár értékű — AT&T technológián alapuló — berendezést fektetett be az SFMT az új cégbe, ezen belül idén egy magyarországi földi elosztóállomás (hub) üzem behelyezését tervezik. Magyarország legnagyobb műholdas távközlési (VSAT) szolgáltatójaként közel 120 állomást — a legnagyobbat a Szerencsejáték Rt számára — installáltak az országban, év végéig további 180 állomás telepítését tervezik.

Sziebig Andrea

Grafikus szövegmegjelenítő

Készítsünk képeskönyvet!

Szinte minden számítógépen találunk valamilyen egyszerű szövegmegjelenítő segédprogramot, amellyel — a DOS barátságtalan TYPE parancsa helyett — a felhasználó belenézhet a szöveges állományokba. A grafikus módban működő The Illustrated Reader a számos karakteres szövegmegjelenítő után új generációt képvisel.

Dokumentumaiba PCX formátumú grafikus ábrákat szúrhatunk be, egyes szövegrészeket vastag betűvel vagy aláhúzottan jeleníthetünk meg. Ez az eszköztár már elegendő lehet arra, hogy szöveges mondandónkat ábrákkal kiegészítve, egyszerűen és gyorsan készítsünk számítógépes képeskönyveket, illusztrált leveleket, képes használati utasításokat stb.

A programocska szolgáltatásai egyszerűek, az Új Alaplap lemezmellékletén is mindig megtalálható TW.EXE-hez hasonlóak. Lapozhatunk oda-vissza, kereshetünk szövegrészre, egy kis menüből állíthatunk a színeken, kinyomtathatjuk a teljes szöveget vagy az aktuális oldalt. Nyomtathatunk grafikusan is, de ettől nem szabad sokat várni, ugyanis csak kétféle nyomtatókategóriát ismer: a lézernyomtatókat és az egyéb printereket. Az utóbbinál Epson-kódokkal 9 tűs szintű nyomtatási képet kapunk.

Beszúrhatunk könyvjelzőt, ekkor a lemezen eltárolódik, hogy hol tartottunk, és a következő indításkor onnan folytathatjuk az olvasást. Végül az aktuális könyvtárból betölthetünk újabb szöveget. Mindezt legkönnyebben egérrel tehetjük, de a billentyűzet is éppoly célszerű.

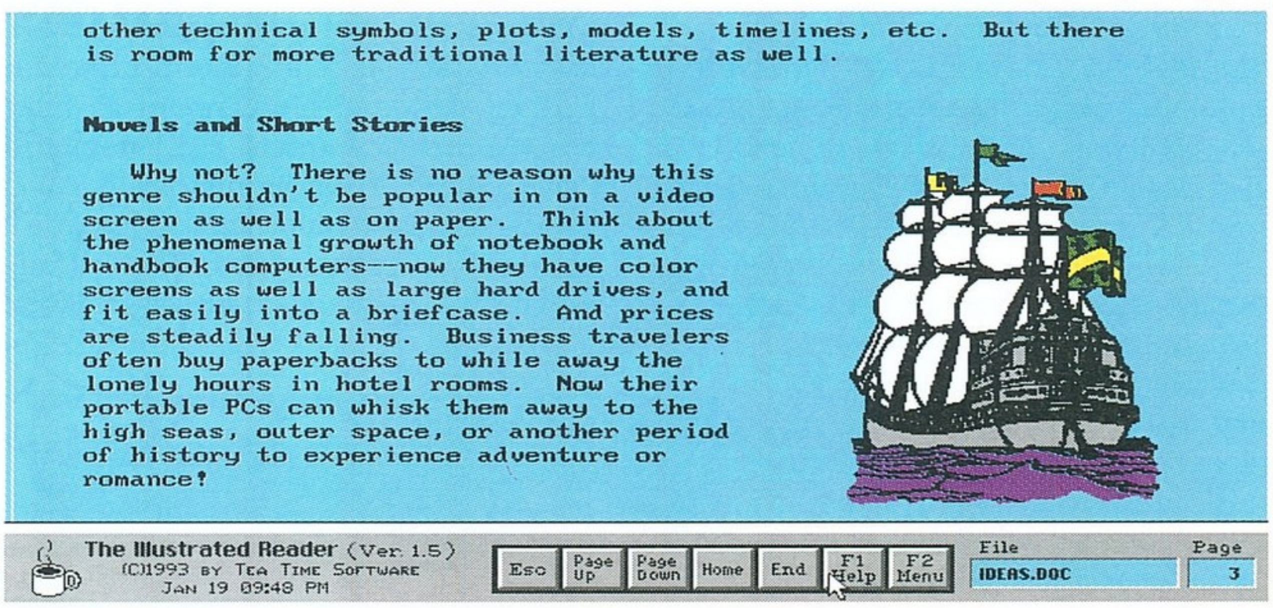
A dokumentumok írása szintén könnyű. Ha egy ASCII formátumú állományt .DOC kiterjesztéssel látunk el, és vigyázunk arra, hogy 75 karakteroszlopnál többet ne tartalmazzon (a többit ugyanis a program szó nélkül lehagyja a képernyőről), már olvasható szöveget kapunk. Itt a teljes ASCII-tábla használható, a magyar karakterek kódjainak eredeti IBM-féle formáját kapjuk vissza.

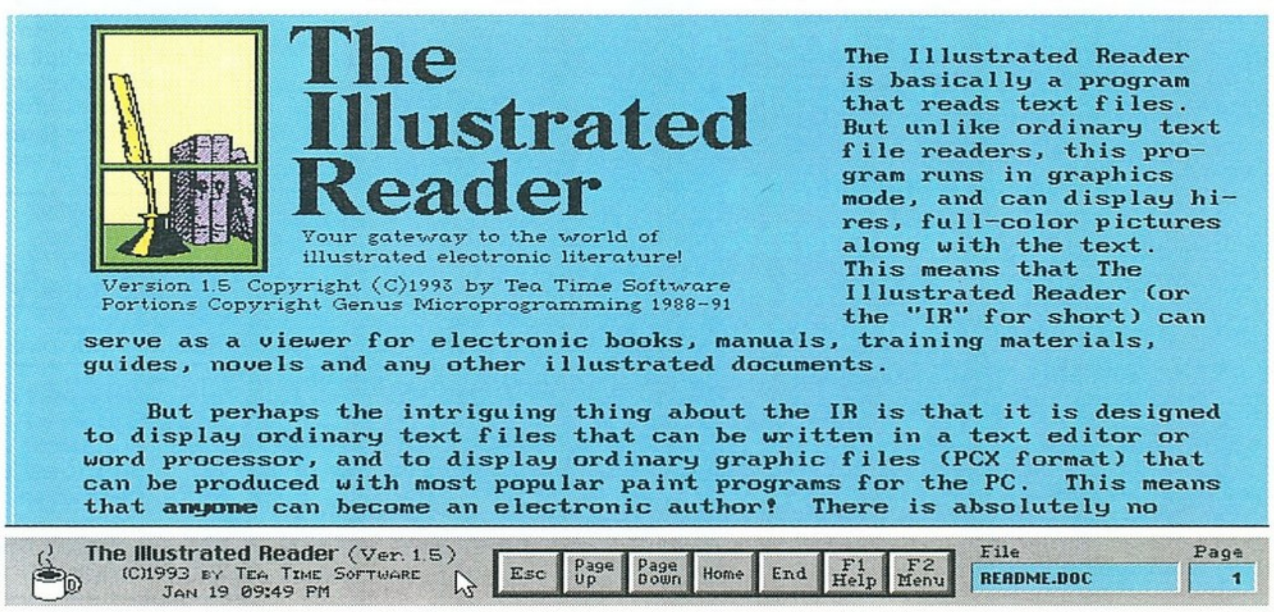
A képek beszúrása némi műszaki szerkesztést igényel. Először meg kell határozni, vagy legalább jól felbecsülni a kép méretét képpontban, azután ki kell számolni, hogy ez hány karakter szélességű és magasságú helyet foglal el a képernyőn. Ki kell kalkulálni azt is, hányadik képernyőoldalra kerül a kép. Minden oldal (az utolsó is) 21 soros kell legyen. A számolgatást azonban a szerzők feleslegessé tették, a dokumentációban mellékelt táblázatokból pillanatok alatt leolvasható a megfelelő érték. (Más kérdés, hogy talán a programba is beépíthettek volna ehhez egy szövegáttördelő automatizmust!)

Ezután a szöveget úgy kell megszerkeszteni, hogy a kép helye üresen maradjon, szóközök (nem tabulátorok) beszúrásával. Végül szükséges egy, a dokumentum nevével megegyező nevű, .FIF kiterjesztésű kis fájl, amelyben a kép nevét, az oldalszámot és az oldalon a kép bal felső sarkának koordinátáit soroljuk fel. Ha a kép az aktuális könyvtárban van, már meg is nézhetjük az eredményt. Hasonlóan készül a vastagon szedett részeket vezérlő .CIF fájl. Itt sajnos nem elég koordinátákat megadni, magát a szöveget is még egyszer le kell írni.

A The Illustrated Reader 16 színű EGA-grafikát használ, ehhez a képeknek is igazodniuk kell. Hibája, hogy a színbeállításokat nem menti el, kicsit lassú, és a lenti egérmenüje sztrájkol, ha nem az aktuális könyvtárból indítjuk a programot.

Eidenpenz József





SOLARSOFT ADATLAP

Lemezszám: 726
Név: The Illustrated Reader v. 1.5
Szerző: Tea Time Software, USA, 1993
Leírás: Grafikus szövegmegjelenítő, képeskönyvkészítő.

makrotrend
ELEKTRONIKAI ÉS SZÁMÍTÁSTECHNIKAI SZÖVETKEZET

1143 Budapest XIV.,
Hungária krt. 65–67.
Tel.: 183-4356 Fax: 163-7888

victron
(holland)

ON-LINE szünetmentes áramforrások
3KVA–30KVA teljesítmény tartományban,
nagy megbízhatóság – alacsony ár.
Szoftverek minden rendszerhez.
MEEI engedély!

Kiváló minőségű hardver-termékeink:

E-NET 16/A AUI+TP, prog.	7 700 forint
E-NET/32 EISA	30 200 forint
Ethernet repeater 2 BNC-s	30 500 forint
Ethernet repeater 2 BNC/2AUI	35 000 forint
Multiport repeater 4 BNC+2AUI	63 000 forint
Ethernet Concentrator/8	35 300 forint
Ethernet Concentrator/16	54 500 forint
Vékony ethernet transceiver	5 100 forint
Vastag ethernet transceiver+csatl.	14 100 forint
Ethernet Local Bridge	167 000 forint
Ethernet Remote Bridge	199 800 forint
Ethernet pocket Adapter	15 800 forint

Viszonteladóknak jelentős kedvezmények.
Áraink az áfát nem tartalmazzák.

INFORMÁCIÓKÉRÉS: A0327 ▲

MAGIC

A NEMZETKÖZI
PROGRAMOZÓI
BAJNOKSÁGON*
A MAGIC
MINDEN ÉVBEN
A HELYÉRE KERÜL

1ST 1ST
1992 1993

Hivatalos magyarországi disztribútor:
ONYX Szoftverház
Telefon: 165-3325, 267-1183
*Durham, Észak-Karolina, USA
Részt vettek többek között: CLIPPER, ACCESS, ORACLE, CLARION, POWERBUILDER, DATAEASE...

SUMY

INFORMÁCIÓKÉRÉS: A0333 ▲

FAN
computer

PROFESSZIONÁLIS SZÁMÍTÓGÉPEK
4 ÉV
GARANCIÁVAL

A/4 COLOR SZKENNEREK
- 1200 és 2400 DPI
- dia-átvilágítás

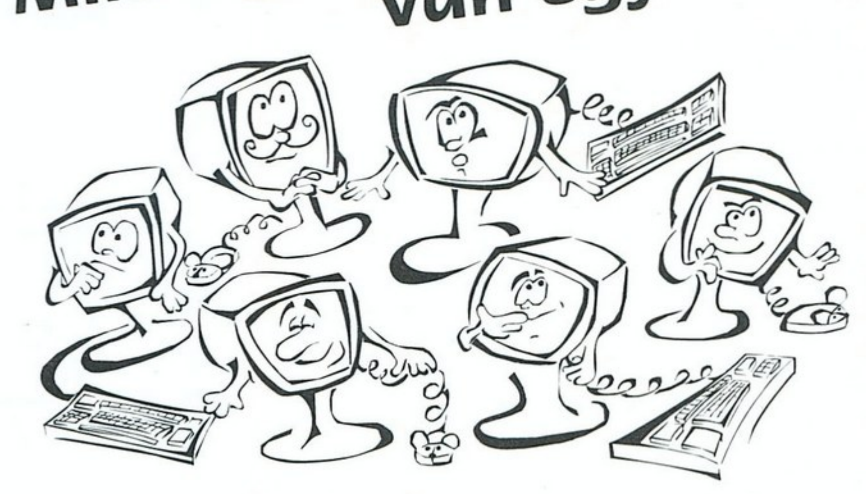
KÉZI SZKENNEREK
DIGITALIZÁLÓ TÁBLÁK
EGEREK, TARTOZÉKOK

ÚJ IRODÁNKBAN,
A KÖRÖNDNÉL IS
1068 Bp. Felső erdősor u. 6.
tel./fax: 141-0799

FAN Electronics Ltd
1118 Bp. Késmárki u. 8.
tel./fax: 185-0813

INFORMÁCIÓKÉRÉS: A0315 ▲

Mindegyiknek van egy álma



Egy személyi számítógépnek is van célja. ▮ Az, hogy Önt még jobban szolgálja, segítse vagy éppen szórakoztassa – a számítógépes kellékek segítségével. ▮
Ezekből a kellékekből a legnagyobb választékot Ön is a PC Kuckó boltokban találja meg. ▮
Íme: floppylemezek, Polaroid termékek, Microsoft szoftverek, egerek, joystickek, monitortartó karok, komplett számítógépek, és... de nem folytatjuk. ▮
Várjuk a PC Kuckó boltokban, és új mintatermünkben is, a Jászai Mari téren. ▮
Jöjjön el, és teljesülni fog az Ön számítógépének álma is. ▮

PC Kuckó A számítástechnika komfortja

Napi információk a TELETEXT 377.oldalán.

Budapest XIII., Jászai M. tér 5. Tel./Fax: 111-5468
Budapest XIII., Tátra (Sallai) u. 8. Tel./Fax: 131-5705
Budapest VII., Thököly út 32. Tel./Fax: 269-7716, 269-7980
Budapest VI., Damjanich u. 23. Tel./Fax: 121-0561
Debrecen, Timár u. 15-19. Tel./Fax: (52) 349-662, 315-563
Debrecen, Batthyány u. 10. Tel./Fax: (52) 312-166
Miskolc, Széchenyi u. 14. Tel./Fax: (46) 356-136
Szeged, Bartók Béla tér 10. Tel./Fax: (62) 322-256

BÓKER REKLÁM

INFORMÁCIÓKÉRÉS: A0311 ▲

Shareware-országi utazások

CD-lemez a magazinban I.

Külföldi kalandozásaink során ezúttal egy olaszországi villámlátogatásra és egy spanyol túrára kalauzoljuk el az olvasót, hogy bemutassunk néhány ottani, shareware- és public domain programokkal foglalkozó számítástechnikai szaklapot, a programok vezető forgalmazóit, azok árait és kínálatát.

Karácsony előtt két nappal Észak-Olaszországban minden csendes. A reggeli órákban sűrű köd üli meg Velencét; a sztrádán szokatlanul gyér a forgalom. A benzinkút mellé épült kisebb bevásárlóközpontban a reggeli sonkás szendvicsek megvásárlása és a pindurka, ám annál kitűnőbb eszpresszókávé elkészülte közben, az újságok és magazinok százai közül kiválogatva a minket érdeklő speciálisakat, jut idő átböngészni: van-e valami érdekes, valami új a számítástechnikában?

Végre előkerül a tizennegyedik éve megjelenő Micro & Personal Computer decemberi száma egy hegymászó szaklap és egy pornómagazin mellől – más semmi. Megvesszük. Este, Franciaországban majd lesz idő átböngészni; azután a felszakadozó ködben, az egyre erősebb napsütésben hajrá, át a Pó-síkságon, irány Genova. Az Appenninek egyik hegyszorosában állunk meg ebédelni, de itt is csalódás ér: az étel hideg, a kisebbik gyerek csak hasábburgonyát hajlandó enni, a Micro & Personal Computeren kívül csak néhány oktató és játékszoftver kapható.

Tétova kezdet

A ligúriai partok százegynéhány alagútján és tucatnyi viaduktján átvezető kényszerű, óránként 130-160-as sebességű száguldozás után egy csendes francia kisvárosban végre nyugodtan át lehet lapozni a „zsákmányt". A tetszetős, de kissé konzervatív havilapban (programlistái 8 pontos betűméretben, bár már lézernyomtatón készülnek!) az MS Works for Windows 3.0, a Novell DOS 7, a WordPerfect 6.0 for Windows és az olasz nyelvű MS FoxPro 2.5 bemutatása után előbukkan egy kétoldalnyi PD- és shareware-rovat: Támadás a vírusok ellen.

A cikk a Datawatch Corporation DOS és Windows alá készített Virex v. 2.91 programját, az amerikai Patricia M. Hoffmann HiperText VSUM nevű, 2317 db vírust leíró katalógusszoftverét, és — az általunk is jól ismert — McAfee-féle, 9.19-es verziószámú VirusScan V108 és Clean-Up V108 programokat ismerteti. Kis keretes anyag ad kurta tájékoztatást a shareware-programok mibenlétéről, a BBS-eknek a terjesztésükben elfoglalt szerepéről, s arról, hogy a lap tizennegyedik oldalán ezzel kapcsolatos hirdetés található.

A megjelölt oldalt fellapozva egy Bologna melletti kis faluba települt cég, a Conti Editore S.p.A. egyoldalas ajánlatát és megrendelő kuponját találjuk, amelyen 12, a Micro & Personal Computer hasábjain az előző másfél évben kéthavonta ismertetett programcsomag, illetve gyűjtemény szerepel. A legfrissebb Antivírus csomag melett megtaláljuk Phil Katz PKZip 2.04-ét, számos windowsos kiegészítést (fontokat, játékokat, segédprogramokat, egy desktop információs menedzsert), és néhány jelöletlen DOS programot. A 3,5 hüvelykes lemezen forgalmazott programok ára egységesen 15 000 líra, vagyis napi árfolyamon kereken 900 HUF (Hungarian Forint).

Nem nagy durranás! — mondhatja erre bárki, de a számítógépgyártás egyik európai fellegvárába (Olivetti!) is betört végre ez a szoftverterjesztési módszer, s talán ebben az évben (vagy a következőkben?) beindul a jól ismert mechanizmus: a kisebb-nagyobb shareware-könyvtárak és a témával foglalkozó szaklapok, végül a BBS-könyvtárak megjelenése.



A kereskedelmi szoftverek árnyékában

Másnap hajtunk tovább. Zsúfolt a program: meg akarjuk nézni Carcassonban Európa egyik legszebb középkori várát, s estére el kell érnünk Andorra-La-Vellába, mert a gyerekeknek karácsonyra — a szállodai szobának a Pireneusok ormain szedett fenyőgallyai alá — egy egész miniállamot ígértünk. Azért Aix-el Provance-ban van annyi idő, hogy átnézzük egy hírlapárus pazar kínálatát...

A számítástechnikai magazinok választéka is meglepően bőséges: mintegy féltucatnyi, szoftverekre, hardverekre és általános felhasználói információkra szakosodott terjedelmes havilap, néhány hetilap (köztük egy IDG-klón), s olcsó szoftverek kézikönyvvel, kartonra fóliázva.

Sajnos az általunk keresett shareware-lapok nincsenek köztük; egyiknek sincs lemezmelléklete, s amennyire a közép-európai általános nyelvtudás francia elemeket is tartalmazó szókincse (igen-nem, jobbra-balra, szabadság-egyenlőség-testvériség stb.) alapján megállapítható: nincs bennük sem PD- vagy shareware-rovat, sem cikk, sem ilyen témájú hirdetés.

Karácsonyi meglepetés

Carcasson az eső ellenére is gyönyörű, a Földközi-tenger menti napfényes utazás után pedig a karácsonyi fényekbe és hóba öltözött Andorrába való megérkezés — egy felhőbe burkolt 2300 méter magas hágón át — egyenesen káprázatos. És Andorra-La-Vellában

egy ünnepi hangulatú, de a pesti lázas bevásárlórohamtól mentes élet fogad. Az utcák nem halnak ki négy óra után, sőt az áruházak és a bevásárlóközpontok este nyolcig, a kisebb üzletek pedig — a mediterrán vidékekre jellemzően — fél tízig-tízig nyitva vannak.

Mindenki kap még a „fa" alá valamit: egy szivárványszínű, bőgő dinoszauruszt (Triceratops Andorrensis), egy Swatch-karórát, egy kalóz-LEGO-t, egy infra fejhallgatót (hogy ne kelljen örökké Axell Rose-t és Bon Jovit hallgatni!), a Beatles CD-n kiadott kék és piros albumát, s végül két gyönyörű spanyol számítástechnikai szaklapot: mindkettőnek van 3,5"-os lemezmelléklete, egyiknek kettő is, a másik mellett pedig egy CD-lemez is található!

Super PC

Karácsony első napja az alvásé, sétáé és az olvasásé. A Super PC még gyerekcipőben járó, 1993 júniusa óta havonta megjelenő magazin — a mintegy háromtucatnyi sport-, szabadidő- és különféle hobbilapot kiadó Grupo Zeta igényes terméke. 150 színes oldalon, drága mellékletekkel (az amerikai AT&T társasághoz tartozó NCR cég hardverajánlatával, a spanyol Asesores Sistamas Multimedia szemet gyönyörködtető, 1200-7000 forintos kereskedelmi programokat ismertető katalógusával) mutatja be a szoftvervilág legújabb fejleményeit.

Az MS DOS 6.0 Backup, a Norton Backup 3.0 és a Central Point Backup for Windows 7.2 összehasonlítása mellett (micsoda kitűnő képernyőfotók!) tág teret szentel a multimédia-szoftvereknek, a grafikus programoknak, az új számítástechnikai jellegű eszközöknek és kiegészítőknek (PC-hez kapcsolható családi zsebnoteszek, távolságmérők, notebook program- és memóriakártyák, botkormányok, egerek, lemeztartók stb.), továbbá a novemberben Madridban megrendezett SIMO 93 szakkiállításnak — és legfőképp a játékoknak.

Komplett WordStar ajándékba

A shareware-témát egy gazdagon illusztrált elméleti cikk, a lemezmelléklet (és annak ismertetése önálló rovatban), valamint régi ismerősünk: a Németországban, Svájcban és Ausztriában néhány éve még a vezető shareware-könyvtárak közé tartozó Soft-Mail A.G. — Barcelonában és Valenciában is önálló irodát fenntartó — spanyolországi leányvállalatának nyolcoldalas fizetett katalógusa képviseli.



A cikk egyfajta bevezetés a shareware-koncepcióba, amely alternatívát jelenthet a szoftverfejlesztőknek és a felhasználóknak egyaránt, a vezető szoftverházak milliárdos — a programok teljesítményét túllihegő, korlátait, hibáit pedig agyonhallgató — „tájékoztatásaihoz", valamint jogi eszközökkel fenyegetőző és zsugorfóliával zsákbamacskává alakított szoftvercsomagjaihoz képest. A Micro & Personal Computer kis keretes anyagához hasonlóan, bár annál terjedelmesebben ismerteti a BBS-ek shareware-terjesztő szerepét, amiből az derül ki, hogy ezek széles körű alkalmazása, miként nálunk, még Spanyolországban is várat magára.

A két lemezmelléklet közül az egyik egy meglepő ajándék: a WordStar 6.0-s komplett csomagja (a lap egyik cikke a 7.0-s verziót öt oldalon keresztül mutatja be!), amely azonban nem shareware, feltehetőleg különleges „sidelicenc" útján megszerzett legális kópia a lap előfizetői és vásárlói számára. Külön érdekessége, hogy a Super PC lemezmelléklet birtoklása a 7-es verzióhoz olcsó upgrade-lehetőséget teremt.

A másik 1,44-es minifloppy 8 valódi shareware-játékot tartalmaz. A DOS alatt futók közül az Apogee Secret Agentje és Major Strykere, valamint az Epic MegaGames Brixe és Jillje olvasóink előtt is ismert. Az egyszerű lövöldözős játékok paródiájának szánt Lamers meglehetősen lapos, a PackManből kifejlesztett ZZT mintájára, a grafikus megjelenítőre írt Snarf útvesztőbeli kincsgyűjtő pedig kefejezetten csúnya. A Windows alá írt Carlton egy érdekes (és nehéz!) pasziánsz-variáns, a külön cikkben ismertett WinRisk viszont igazi gyöngyszem a stratégiai játékok kedvelőinek.

A Soft-Mail A.G. hirdetése is számos friss információt tartalmaz. A cég forródrótja 365 napon át, 24 órában áll az érdeklődők — megrendelők, reklamálók és tanácstalanok — rendelkezésére.

Grafikus és játékszoftverei között sok a régebbi vagy újabb ismerős (DanCAD 3D, NeoPaint, DrumBluster, Duke Nukem stb.), ám a vegyes felhasználói programok és fejlesztő eszközök már felerészben spanyol termékek (Visual Clipper, Generador de Programas en dBASE, Editor y Tutorial de Ficheros Batch). Az ügyviteli szoftvereket pedig kizárólag spanyol programozók jegyzik (Luis J. Jañez y Rafael Espinola, Santiago Ruiz Alhambra, Ms. Gloria Rodriguez és mások).

Mindez talán nem véletlen, hiszen a nemzetközi kereskedelmi életben a második leggyakrabban használt nyelv az angol után a spanyol, a nagy kereskedelmi szoftverházak, de kisebb shareware-cégek is egyre gyakrabban jelentetik meg programjaik spanyol vagy spanyolul is használható változatát.

A SoftMail szoftverek ára viszonylag magas: lemezenként 980 pezeta (686 forint). Hat lemez vásárlása esetén a hatodik ingyen van, 12 lemez megvételekor pedig közel 45 százalékos az engedmény.

Lemeztartó az előfizetőknek

A Super PC példányonkénti ára 750 pezeta (525 forint), éves előfizetési díja tizenkét számra 7200 pezeta (5300 forint). Azoknak, akik 9000 pezetát (6300 forint) fizetnek, a kiadó egy 80 darabos, emblémázott és kulcsos lemeztartót küld „ajándékba". Az előfizetés történhet VISA-vagy American Express-kártyával, illetve spanyol postai folyószámláról. A külföldi előfizetésekről a lap nem tartalmaz információt.

A Super PC előfizethető az Ediciones Mensuales, S.A. kiadóvállalatnál (Apartado de Correos, 1103 FD, 28080 MADRID; ügyintéző: Charo Muñoz; telefon: (91) 578 15 72 /361 vagy 365-ös mellék). Terjeszti a Distribuciones Periódicas, S.A. vállalat (Bailén 84, 08009 BARCELONA; telefon: (93) 484 66 34).

Vékony Tamás

SolarSoft programkönyvtár

Update, upgrade

CDMaster v. 7.12a

A kompakt diszkekhez (CD-khez) készült lemeznyilvántartó program (245) Clipperben íródott, és ezért nagyon gyors. Egyszerű az adatbevitele: tartalom, szerzők, dalcímek, a művészek nevei. Ezek rögzíthetők, a visszakeresés eszerint lehetséges. Nyomtatott katalógust (beköthetőt vagy kártyalap méretűt), valamint a CD-lemez elejére és oldalára ragasztható címkéket is készíthetünk vele. Könnyen installálható, átszínezhető, barátságos program.

Kimenete kerülhet képernyőre, nyomtatóra, állományba. Mátrix- és lézernyomtatókat támogat. Igen rugalmas, például használhatók a felhasználó által definiált mezők. Négy kritériumot adhatunk meg a lekérdezéseknél.

Korlátozása, hogy a 6.40-esnél régebbi verziókról való áttéréskor (a SolarSoft 1990-ben leragadt az 1.2-nél!) az adatállományok nem konvertálhatók. A szerző csak a regisztrált felhasználóknak vállalja ezek átalakítását. Az újabb változatok adatainak konvertálása a program menüjéből történik. Ne installáljuk meggondolatlanul az új verziót, készítsünk biztonsági mentést adatainkról!

Javításai a 6.2-es verzió óta:

— Javított listakezelés a Delete és Update menüpontoknál, ami kevesebb gépelést eredményez.

— A menük és a háttér javítása több helyen.

— Az egértámogatás bevezetése.

— Környezetérzékeny help, amelynek hívása végre a szokásos F1-re érvényesül.

— Lézernyomtatók támogatása, egyes esetekben félkövér nyomtatás.

— A színek átdefiniálásának lehetőségei bővültek.

— Kilépésnél biztonsági mentés.

— Korlátlan számú címbejegyzés lehet egy lemeznél.

— Újabb, a felhasználó által definiálható mezők hozzáadása (max. 3).

— Kétféle méretű katalóguskártya nyomtatható.

— A nemzeti dátum- és időformátumok támogatása.

DanCAD 3-D v. 2.5

Az USA-beli Daniel H. Hudgins az elmúlt évek során több száz kisebb javítással alaposan átdolgozta a CAD/CAE-alkalmazásokhoz készült professzionális mérnöki tervezőrendszerét (#39/1, 2). Az egértámogatás menükre kiterjesztésével sokat javult a program fehasználói felülete, de azért megmaradt a parancsok indításának lehetősége kezdőbetűkkel is: (Z)oom, (R)otate, (F)lip, (U)ndo, (M)agnifity, (R)ectangle, (D)raw, (W)rite, (2)D, (3)D stb.

A rajzolás és a megjelenítés eszköztára számos új paranccsal, illetve móddal bővült, a rajzok megjelenítését támogató makrók futási sebessége pedig nyolcszorosára nőtt. A feliratozást már mini-editor és fonttervező is támogatja, amely utóbbi végre lehetővé teszi a nemzeti karakterek használatát is. S bár a DanCAD-nek csak a regisztrált változatához jár a DanMovie és a DanGraph program, a shareware-változatba is beépült egy ezekhez hasonló animációs lehetőség.

Az új nyomtatómeghajtók (összesen 59 különféle) révén a program a lézer-, a mátrix- és a tintasugaras nyomtatók, illetve a plotterek számtalan típusát, valamint a PostScript-formátumot támogatja. Négy demója, javított helpje és részletes dokumentációja pedig a kezdő és a gyakorlott felhasználót egyaránt.

PC-Write Advanced Level v. 4.01

A Quicksoft szövegszerkesztő és feldolgozó programjának (#54/1, 2, 3) kezdeti verziója az első nagysikerű shareware-program volt az USA-ban és az egész világon. A SolarSoftból megismert 2.71-es és a későbbi, 3.05-ös verziója után szerzője, Bob Wallace, akit a shareware atyjának is neveznek, a 3.x sorozatot PC-Write néven nem fejlesztette tovább, helyette három új programot, illetve programváltozatot jelentett be.

A ProLite (#506) kis memóriaigénye miatt jól használható régebbi vagy hordozható gépeken. A PC-Write Standard Level 2.0 tartalmazza a programcsoport fontosabb szolgáltatásait, s regisztrált verziója nagyon olcsó. A PC-Write Advanced Level pedig kibővített, a gyakorlott felhasználók számára beépített lehetőségeket ad, de értelemszerűen a korábbi és az egyszerűbb verziók, így például a Standard Level összes funkcióját is.

Az Advanced Level installálásánál is kérhetünk szűkített installálást. Ekkor karcsúbb lesz a programunk, ám egyes dolgokról (helyesírás-ellenőrző, help) le kell mondanunk. Floppyra is telepíthető.

Ha van elég (mintegy 1,5 Mbájt) szabad lemezterületünk, érdemes a teljes telepítést kérni. Az ADAPT menüpont segítségével is beállíthatjuk ugyanis, hogy a szolgáltatások mely körére tartunk igényt. Például kezdhetünk a szűkített szolgáltatáscsomaggal, majd igényeink vagy tudásunk fejlődésével egyszerűen a menüből feljebb léphetünk egy szinttel.

A szoftver nyújtja az összes klasszikus szövegszerkesztési szolgáltatást. Ami általában ezeken túlmutat:

— Angol helyesírás-ellenőrző.

— Legördülő menük.

— Egérkezelés.

— Környezetfüggő help.

— Billentyűkombinációk a leggyakoribb funkciók automatizálására.

— Tanulóprogram.

— Elválasztás.

— Levélminták.

— Max. 10 ablak kezelése.

— Több hasáb használható.

— CGA/MGA/EGA/VGA oldalmegjelenítés nyomtatás előtt.

— Tartalomjegyzék-generálás.

— Automatikus oldaltördelés.

— Több állomány tartalmából történő indexkészítés.

— WordPerfect, DCA és ASCII állományok importja.

— Makrók, automatikus makrórögzítés.

— Több mint 900-féle printer támogatása.

— 200-nál több beállítási lehetőség és szolgáltatás.

— Kinyomtatható kézikönyv.

A program egyszerűen az önkicsomagoló állományok kibontásával is telepíthető. Ekkor a CUSTOM.EXE-vel állíthatók be az ED.DEF állomány értékei. Ha mégis a mellékelt installációs rutint kívánjuk használni, másoljuk be az önkibontó állományokat egy üres alkönyvtárba, indítsuk el őket, majd a kicsomagolás után indítsuk az INSTALL.EXE-t. A regisztrált verzióban szinonimaszótár is van.

Játékvilág

Háborús kockázatok

Valószínűleg sokan ismerik és szeretik a Rizikó nevű társasjátékot, amely az elmúlt évtized során olyan klasszikusok mellé emelkedett, mint a Dáma, a Ki nevet a végén, vagy éppen a Monopoly. A DOS alatt futó számítógépes változat nem aratott túl nagy sikert, mivel VGA-grafikára (tábornokok arcképe, vonuló hadoszlopok, lobogó zászlók, tankcsata és szuronyroham) koncentrál, ideológiai mázzal önti le az egészet, s ráadásul a kezelése is nehézkes. A WinRisk azonban visszaadja az eredeti játék tiszta logikai szépségét: a rövid és a hosszú távú tervezés izgalmát, a merész taktikai és stratégiai kombinációk megvalósulásának örömét.

A WinRisk csupán néhány apróságban tér el az eredeti táblás játéktól:

— Térképén a kontinensek egységesen szürkék induláskor, majd mindig felveszik annak a játékosnak a színét, amelyik utoljára elfoglalta.

— Néhány terület elnevezése más (például Kamcsatkát Koryaknak hívják).

— Csak három játékos játszhatja (színük piros, kék és zöld).

— A (nem látható) jutalomkártyák csak a gyalogos-lovas-ágyú jelölésnek felelnek meg, de nem tartalmazzák az országokat, és célkártyák sincsenek. Induláskor mindhárom játékos 21 hadsereget tehet fel tetszése szerinti csoportosításban, s az egész világ elfoglalása a tét.

Állóháború a frontokon

A játék a File menü New Game parancsával indítható. Ilyenkor megjelenik egy kis párbeszédablak, amelyben megadható a három játékos neve, s az, hogy melyik szerepet játssza a számítógép. Alapértelmezésben mi vagyunk a pirosak, s a gép képviseli a kék, illetve zöld színeket. Ugyanebben a menüben található az End Game parancs is, amellyel megadhatjuk magunkat, vagy véget vethetünk az órák óta tartó állóháború kilátástalan unalmának, és az Exit. Kár, hogy az állás nem menthető el, talán egy későbbi változatban gondol rá a szerző.

Várható a játék továbbfejlesztése — ez két dologból is látható. Az egyik az egyelőre még bekapcsolhatatlan Comm menü, amely minden bizonnyal a modemen keresztül megvívható elektronikus csatákra ad majd lehetőséget, a másik pedig a képernyő jobb alsó sarkában megjelenő, végig 00:00 nyugalmi helyzetű óra.

Az Option menü egyszerű: kétféle kártyakészlet kérhető, az alsó, 4, 6, 8, 10, 12, 16... hadsereget adó, és az alapértelmezés szerinti felső, 4, 6, 8, 10, 12, 15, 20, 25... sorozat. Itt korlátozhatjuk az egy országra feltehető hadseregek számát is tizenkettőben, ami kiegyensúlyozottabb, kevesebb meglepetést és kisebb kockázatokat tartalmazó, ám ugyanakkor kevésbé izgalmas játékot eredményez. Kezdőknek, botcsinálta hadvezéreknek azonban mindenképpen javasolt.

Áttörés a szárnyon

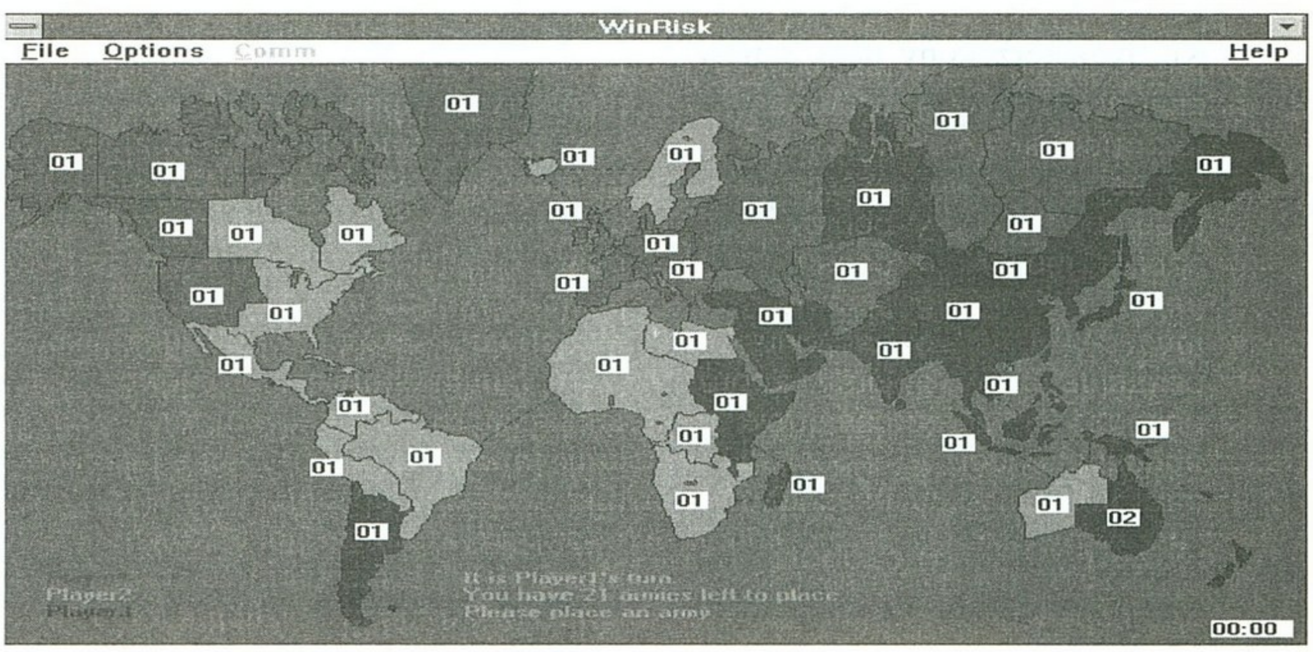
Ha a gép ellen játszunk, nem árt, ha tudjuk: az 1.10-es verzió óta nem egy nyavalyás véletlenszám-generátor az ellenfelünk, hanem egy mind kifinomultabb hadművészeti képességekkel felruházott stratégiai algoritmus, amely védi a szárnyakat, képes az ellenség gyenge pontjainak meghatározására, és ennek megfelelően a csapásmérés fő- és mellékirányának kijelölésére, a többfrontos harcra, és a bekerítésből való kitörésre. Eszköztárába tartozik az átkaroló hadművelet mellett a felmorzsolás, a meglepetésszerű támadás és a gerillahadviselés is.

A véletlen sem hiányzik azonban a játékból: az elfoglalt országokért kapott kártyákból előre kiszámíthatatlan, hogy melyik játékos vált be hármat legközelebb, s ilyenkor nem őrzi meg a három egyforma vagy három különböző kártyán felül megmaradtakat — újra kell kezdenie a gyűjtögetést.

Mindezek alapján nem meglepő, hogy még gyakorlott játékosokkal is elő-előfordul, hogy színük hipp-hopp eltűnik a térképről, s már csak a két gépi játékos gigászi csatájának végtelen ciklusát figyelhetik, amelyből csak a Ctrl+Alt+Del segítségével szabadulhatnak.

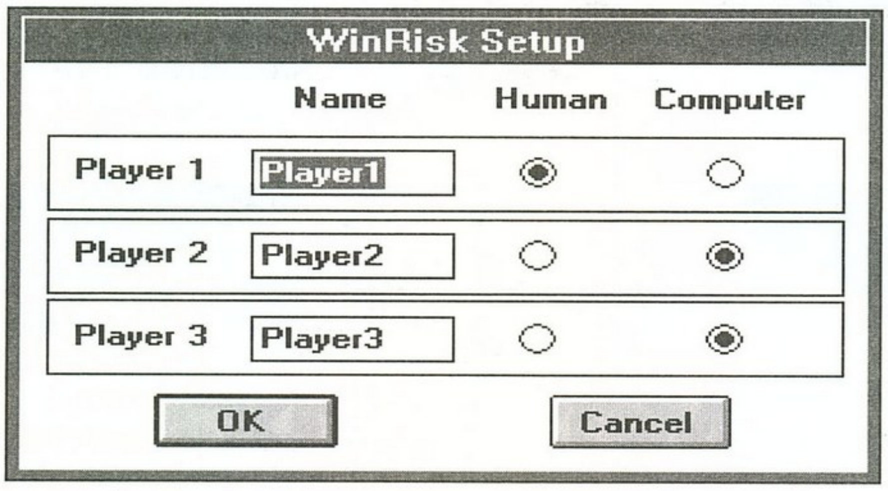
Átcsoportosítás vagy visszavonulás

A jó hadvezér tudja, hogy mikor és hová kell átcsoportosítania erőit, akár mert fel kell adnia egy hosszabb távon tarthatatlan hadállást, akár azért, mert egy csapásmérő csoportosítást von



össze. A WinRiskben ez is eltér egy kissé a táblás változattól. Utóbbival szemben nem vihetjük át a harcoló hadseregeket az általunk megszállt terület tetszőleges pontjára, csupán a szomszédos országba. Ezt az egér jobb oldali gombjának lenyomásával kezdeményezhetjük, s miként az eredeti játékban, az átcsoportosítás befejeztével itt is a következő játékoson a sor.

Az egérvezérlés egyébként egyszerű. A bal oldali gombbal rakhatjuk le, vagy csoportosíthatjuk át egyesével a hadseregeket, és ezzel jelölhetjük ki azt is, hogy honnan hová támadunk. A jobb oldali gomb az átcsoportosítás funkció bekapcsolásán túl arra is lehetőséget ad, hogy ötösével tegyünk le vagy csoportosítsunk át hadseregeket. A Help szerint a 3 gombos egérrel tízes egységek is letehetők (ezt azonban nem állt módunkban kipróbálni!).



Vereség egy egér miatt

A háromgombos egérre írt program Passz funkciója a középső gombra kattintással érhető el. Kétgomboson ezt az egyidejű kopogásnak kellene végrehajtania — ez azonban MS Mouse-on nem működik. Helyette a kényszerű átcsoportosítás javasolható, vagy pedig az egérkezeléshez hozzáidomult játékos számára kissé szokatlan és kényelmetlen billentyűhasználat: a Föl kurzorvezérlő gomb lenyomása.

Feltehetőleg programozási hiba, hogy India térképén a hadseregek számát kijelző mező körül az egérrel hiába kopogtatunk. Csak a szubkontinens határvonalainak környékén érzékeli kattintásunkat a gép. Más országoknak is vannak kisebb „érzéketlen" területei, ezek azonban nem okoznak a csata lázában égve olyan bosszúságot, mint India, ahol többszöri kattintás után még mindig nem történik semmi.

Ugyancsak az egérkezeléshez tartozik, hogy könnyű mellékattintani. A néha előforduló Csal a gép! felkiáltások alapja is a hibás kattintás, a tervezettnél eggyel többször, vagy rossz oldalon lenyomott gomb. Mivel a játék menüjében nincs Undo funkció, az egér miatt megsemmisülhet néhány hadsereg, elveszhet egy ország, egy teljes kontinens, de akár az egész háború.

SOLARSOFT ADATLAP

Lemezszám: 741
Név: WinRisk v. 2.11
Szerző: Steve Stancliff, USA, 1992
Leírás: Rizikó-társasjáték három játékosra.
Konfiguráció: i286 vagy nagyobb, 640x480-as VGA képfelbontás, egér.

INFORMÁCIÓKÉRÉS: A0305 ▶

INFORMÁCIÓKÉRÉS: A0314 ▶

INFORMÁCIÓKÉRÉS: A0324 ▶

INFORMÁCIÓKÉRÉS: A0309 ▶

Dinamikus programozás — élőben

Jelsorozatok egymáshoz illesztése

Ígéretünknek teszünk eleget, amikor most egy nagyobb méretű feladat példáján ismertetjük, hogyan alkalmazzák ma a dinamikus programozás szellemesen vonzó elvét a gyakorlatban.

Az egyik legjobban bevált alkalmazási területe a dinamikus programozásnak olyan feladatok megoldása, ahol jelsorozatok közötti összehasonlítással, ezek közös részeinek megtalálásával, illetőleg bennük bizonyos minták alapján azonos részek keresésével kapcsolatos az alapprobléma. Ilyen feladat például a genetikus kódok elemzése. Gyakran kilométer hosszúságú DNS-kódkombinációkat kell összevetni egymással, nem is tudva előre, hogy milyen kódmintázatokban fogunk egyezést találni. A klasszikus keresési eljárások ezért vagy egyáltalán nem, vagy csak nagyon kis hatékonysággal alkalmazhatók.

Újonnan felfutóban lévő alkalmazás, amely azonban módszereiben sok hasonlóságot mutat az előző feladattal, beszédfolyamatok vizsgálata különböző nyelvi szinteken. Ismeretes, hogy Japánban évek óta folyik egy nagyszabású kutatási téma a legnagyobb telefontársaságok összefogásával. A kutatás elsődleges célja többnyelvű telefonbeszélgetések szinkron tolmácsolásának számítógépes megoldása. Érdemes megjegyezni, hogy az online megoldás alkalmazása nehézségei ellenére nagy könnyebbséget is jelenthet. Ha ugyanis a partner nem érti meg a fordított szöveget — mert félrefordítás történt, vagy mert a rendszer maga felismeri, hogy nem tud elbánni a szöveggel, és például az eredeti szöveget közvetíti a fordítás helyett —, akkor a hallgató közvetlenül rákérdezhet. Így a beszélőnek lehetősége van akár világosabban tagolva, akár más szavakkal megismételni, bővebben kifejteni mondanivalóját.

Rendkívül összetett probléma

Nyilvánvaló, hogy több, egymást követő nyelvi szinten kell elemzést és transzformációt végezni. A többszintű megoldás jellemzője, hogy a megelőző szint outputja adja a következő szint inputját, és így tovább, ami természetesen magában hordozza az esetleges hibás döntések következményeinek felerősödését, és végső soron időnkénti lokális kudarcok fellépését. Az egyes szintek az alkalmazott nyelvészeti egységek jellege szerint különíthetők el. Először hangtani elemzésre van szükség a beszéd jelentéshordozó elemeinek felismerése érdekében. Ahol lehet, természetesen itt is maximális mértékben ki lehet és ki is kell használni a környezeti adottságokat és az ismereti hátteret, a feladat tehát erősen környezetfüggő a szó tágabb értelmében. A már felismert beszédelemek éppúgy jó kapaszkodót nyújthatnak, mint a még elemzetlen beszédfolyamban felbukkanó jegyek orientáló szűrője.

A hangtani elemzés outputja képezi az inputját annak a szintnek, amely már a grammatikai és lexikai morfémák felismerésére és értelmes egymáshoz illesztésére irányul. Ezen a szinten különösen nagy jelentősége van egy nagyméretű, de ügyesen mozgatott, intelligensen kezelt adatbázisnak. Ennek az a szerepe, hogy a közvetlenül felismert információmorzsáknak mint kulcsoknak a segítségével további plauzibilis információkkal egészítse ki a rendelkezésre álló, szinte minden esetben hiányos információegyütteseket.

Az előhalászott információk feldolgozásában a fő problémát az jelenti, hogy miként tudjuk biztosítani a különböző „információszigetek" összeegyeztethetőségének vizsgálatát. A vizsgálat és az együttes előfordulás lehetőségeinek mérlegelése alapján kiszűrjük és elvetjük a selejtnek bizonyult ballasztot (vagy teljes bizonyossággal, vagy valószínűségi módszerek alkalmazásával — feltéve, hogy alá kerülnek egy valószínűségi küszöbnek), és megerősítjük a megmaradt hasznos információk megbízhatósági értékét. E szint outputja már egy komplett szintaktikai, vagy akár szintaktiko-szemantikai analízis lehet.

Innen már nem követjük tovább a probléma bonyolult megoldási útját. Az mindenesetre nyilvánvalónak látszik, hogy jelsorozatok egymáshoz illesztése sok helyen szerepet kap a többszintű elemzés folyamatában, és az is világosan látszik, hogy maguknak a jelsorozatoknak az elemei a legkülönbözőbb minőségű egységek lehetnek.

Különböző nyelvű szövegekkel

A gépi adathordozón elérhető szövegek (vagy akár mesterségesen előállított jelsorozatok) hatalmas növekedésével megnőtt a számuk és jelentőségük egyes alkalmazásoknak. Ezek célja az egymással szoros korrelációban lévő, egymásnak megfelelő részek összeillesztése az esetleg egészen más formában (például más nyelven) megadott szövegekből (vagy szövegnek tekinthető mesterséges jelsorozatokból).

Egyik konkrét és jól ismert példája az ilyen tömegesen elérhető kétnyelvű szövegeknek a kanadai parlament folyamatosan megjelentetett "Hansardjainak" jelenleg kb. 90 millió szavas korpusza, angol és francia nyelven. (Hansardnak hívják angol nyelvterületeken a parlament jegyzőkönyveinek hivatalosan megjelentetett kiadványait Luke Hansard brit nyomdász neve után, aki 1774-ben elkezdte publikálni az angol parlamenti jelentéseket, s utódai egészen a múlt század végéig folytatták e nemes hagyományt.) Az említett korpusz elérhető például a nemzetközi Számítógépes Nyelvészeti Szövetségen keresztül (Association for Computational Linguistics). A közelmúltban létrehozott szervezeti egységüknek (Data Collection Initiative) kimondottan az a célja, hogy egyrészt koordinálja, másrészt közvetlenül elősegítse a szöveges információk elemzésével foglalkozó kutatók és kutatóintézetek adatellátását.

E kutatásoknak természetesen csak eszköze, nem célja a kanadai parlamentben elhangzott beszédek ízekre szedése.

Ezek a szövegek azért alkalmasabbak a kutatásokra, mint — mondjuk — Shakespeare drámáinak különböző nyelveken elérhető, bizonyára páratlanul értékes fordításai, mert Kanadában rendkívül komolyan veszik a többnyelvűséget, így igen nagy gondot fordítanak a parlamenti szövegek gondosan szerkesztett, tökéletesen szöveghű, az eredetivel egyenértékű fordítására is. (Az eredeti egyébként hol francia, hol angol nyelven hangzik el.) Az sem véletlen, hogy az első üzemszerűen működő számítógépes rendszer is Kanadában született meg, a sűrűn kiadott meteorológiai jelentések azonnali kétnyelvű előállításának megkönnyítésére.

Alapvető nehézségek

E kétnyelvű szövegek érdemi elemzésének első lépésében mondatról mondatra meg kell állapítani, hogy az egyik nyelven megtalálható mondatnak melyik a megfelelője a másik nyelven. A probléma abból keletkezik, hogy egyrészt kihagyott vagy beszúrt mondatok is előfordulnak a szövegben a leghívebb fordítások esetén is, másrészt időnként két vagy esetleg több mondat összeolvadásából keletkezik egy, vagy megfordítva: szétvág a fordító egyetlen mondatot több önálló részre. Kevésbé gyakori jelenség az, amikor a mondatok sorrendje felcserélődik a fordításban (lehet, hogy pontosan az értelmi összefüggés hűbb visszaadása érdekében — gondoljunk az élő nyelv kisebb-nagyobb bakugrásaira), vagy több mondat együttes értelmezéséből keletkezik rögtön több mondat.

E feladat megoldására szolgáló algoritmusnak tehát eleve nem csekély nehézségekkel kell megküzdenie. Az alapvető gondot a két nyelv építkező elemeinek (szavainak) szótárba sem foglalható eltérései okozzák, ahol ráadásul ezeknek az elemeknek a száma és sorrendje is csak kivételképpen egyezik meg. Ugyanakkor hihetetlen erőpocsékolást jelenthet egy olyan mondathatár-illesztési módszer, amely az egymásnak megfeleltethető szavak összekeresgélésén alapul, s ennek alapján jut el mondatok megfeleltetéseinek megállapításához. A fordított út sokkal életszerűbbnek tűnik a feldolgozás automatizálása során: érdemes előbb viszonylag gyors (ha nem is hibátlan) eljárást alkalmazni a mondatok egymáshoz illesztésére, majd utána, ennek eredményei alapján lehet tovább folytatni az elemzést az egymásnak megfeleltethető szavak kibányászására, gyakoriságuk és környezeti feltételeik mélyebb elemzésére.

Igen hasznos további terméke lehet egy ilyen mélyebb, lexikai szintű elemzésnek a két nyelv szavainak valószínűségi súlyokkal ellátott szótárszerű adatbázisa. Az IBM egyik kutatóintézetének munkatársai (P. Rrown, J. Cocke, S. Della Pietra és mások) már jó három évvel ezelőtt mutattak be mintákat egy ilyen gyakorisági „fordító szótárból". Szemléltetésül:

Angol	Francia	Valószínűség
the	le	0,610
the	la	0,178
the	l'	0,083
the	les	0,023
the	ce	0,013
the	il	0,012
the	de	0,009
the	à	0,007
the	que	0,007

Egy másik típusú termék (természetesen ez sem végtermék) lehet az ún. konkordancia-szótárak köre — ezek különböző szövegkörnyezetekbe ágyazva mutatják be az egyes szavak használatát. A konkordancia-szótárok fejlettebb változata már gyakorisági mutatókat is tartalmaz, illetve tipológiai rendszerezésre is képes a szókörnyezetek elemzésének megkönnyítése céljából.

Az IBM kutatói egyébként is igen nagy fantáziát látnak a statisztikai alapú számítógépes fordítás módszereinek kidolgozásában. Valószínűnek látszik persze, hogy egyedül ez a megközelítés minőségileg átléphatatlan korlátokba ütközik, a minőségi elemzés más módszereivel kombinálva azonban nem látszik reménytelennek az ügy. (Kétségtelen, hogy a gyakoriságnak mint prioritási szempontnak a figyelembevétele elengedhetetlen a hatékony fordítórendszerek kifejlesztéséhez.)

Jó, jó, de hogyan?

Az alábbiakban nagy vonalakban funkcionális ismertetését adjuk egy olyan modulnak, amely a dinamikus programozás elvei alapján egymásnak megfeleltethető mondatok egymáshoz illesztését végzi el. A program C nyelven készült, számítástechnikai kidolgozója M. D. Riley, nyelvészeti és matematikai inspirátorai W. A. Gale és K. W. Church, mindketten az AT&T Bell Laboratórium kutatói. A programból lemezmellékletünkön részleteket mutatunk be, különös tekintettel azokra a megoldásokra, amelyek esetleg tapasztalt C-programozóink számára sem feltétlenül kézenfekvőek.

A számításokhoz felhasznált alapinformáció a mondatok hossza. Gyakorlatilag ez azt jelenti, hogy a mondat tartalmi mondanivalójának akár csak felületes (például egyes lexikai elemekre kiterjedő) vizsgálata helyett ennek egy számszerű mutatóját veszi alapul.

Monoton, azonos szerkezeti felépítésű közlemények egymásutánja esetén ez a módszer nyilván nem lenne alkalmazható. Még az is kérdéses, hogy nagyon eltérő szerkezetű nyelvek esetében (például amikor az egyik nyelv kínai, a másik valamilyen európai nyelv) mennyire válna be. Tipikus (nem mesterségesen kreált) szövegépítkezés esetén azonban jól kimutatható, hogy a legkülönbözőbb nyelveknél az esetek többségében megőrződik a „hosszabb"—„rövidebb" reláció a mondatban kifejeződő azonos mondanivaló szavakban mért hosszát illetően. Különböző nyelveket egybevetve azt tapasztaljuk, hogy vannak bőbeszédűbb, és vannak tömörebben fogalmazó nyelvek, de nagyjából még ezek között is megteremthető az összhang valamilyen konstans szorzó alkalmazásával.

Érdekes megfigyelni, hogy létezik egy nyelvfüggetlen tendencia is a nyelvekben: a hosszabb és rövidebb mondatok váltakozásában bizonyos ritmika mutatható ki. E jelenség magyarázatát bízzuk az esztétikára, számunkra most csak annyiban lényeges, hogy ez a jelenség is erősíti ennek a látszólag teljesen üres mutatónak a használhatóságát.

Függvények és kaptárak

Igen szellemes az a módszer, ahogyan Riley megoldja a dinamikus programozáshoz szükséges adathalmaz több szempontú válogatását és csoportos kiértékelését. Életre kelnek, sőt önálló életet kezdenek élni függvényként definiált tömbök. Logikai függvényként viselkednek kitöltött és kitöltetlen (0 értékű) argumentumok, és sokarcúvá változtatják „anyafüggvényüket". Az algoritmus kaptárakba gyűjti a hatarcú függvényből csordogáló mézcseppeket, majd pörgetni kezdi a mézet. „Ki mit tud"-szerű helycserés ringlispíllel válogatja ki a nyertest, majd betájolja az optimális irányokat. Ezután valószínűségi súlyozással apró léptekben döngicsél visszafelé, óvatosan követve mindenütt az optimális utat.

A súlyozáshoz paraméterként felhasználja azokat a gyakorisági értékeket is, amelyeket ezerháromszázvalahány angol és francia mondat kézi megfeleltetéséből összegyűjtött. Kiderült ebből, hogy a minta szerint az esetek 89 százalékában (1167 esetben) egy angol mondatnak egy francia mondat felelt meg, 13 esetben (0,99 %) vagy az angol, vagy a francia mondat kihullott (törlés/beszúrás), 117 esetben (8,9 %) vagy két angol mondatnak egy francia, vagy megfordítva, egy angolnak két francia, végül 15 esetben két-két francia mondat tudott kielégíteni két-két angolt (1,1 %). Feltéve, hogy ezek az arányok másutt is ilyenek, meghatározható a deviáns viselkedésű párocskák előfordulási valószínűségének aránya a normális viselkedésűekhez képest, s ennek *e* alapú logaritmusa „devianciaadóként" felhasználható az algoritmusban.

Ez a büntetésként kirótt adó ahhoz hasonlóan épül bele a számításokba, ahogyan régebben felhozott példáinkban az út fajlagos költségeivel súlyozni lehetett az út hosszát. Az utakkal való analógia továbbvitelével az itt alkalmazott megoldást úgy képzelhetjük el szemléletesen, hogy van egy „majdnem-négyzet" terünk (nevezzük például FRANGOL-nak), amelynél az egyik oldal a francia szavak számával arányos, a másik az angollal. Ha mindenütt pontosan egy francia szó felelne meg egy angol szónak, akkor lenne ez a tér négyzet alakú. Képzeljük magunkat a „majdnem-négyzet" jobb felső sarkába. Ebből a sarokból ereszkedünk visszafelé az algoritmus befejező részében oly módon, hogy vagy egy lépést balra léphetünk és egyet lefelé (ez a normális), vagy csak egyet balra, vagy csak egyet lefelé, vagy kettőt balra, egyet le, csak egyet balra, kettőt le, vagy kettőt balra, kettőt lefelé. Ez a hatféle tánclépés felel meg a hatarcú függvény hatféle viselkedésének. A befejező részben azonban már nincs választás, ilyenkor már determinálva van az út, hogy mennyit kell lépnünk az egyes koordináták irányába.

Akit bővebben érdekel a probléma, mélyedjen el a lemezmellékleten a program részleteinek tanulmányozásában.

Vargha Dénes

Adalékok a C++-hoz

„Objektívünk" előtt: az I/O funkciók I.

A hagyományos C és a C++ input/output lehetőségei több ponton is lényegesen eltérnek egymástól. Mielőtt belemerülnénk a különbözőségek taglalásába, le kell szögezni: a C++ felülről kompatibilis a C-vel, így az ott megszokott beviteli/kiviteli eszközök változatlanul használhatóak C++ környezetben is (nyugodtan alkalmazhatjuk például a printf( ), scanf( ) függvényeket, természetesen a stdio.h fejléc fájl beszerkesztését követően). A hagyományos input/output eszközök azonban már nem képesek kielégíteni az objektumok használatával kapcsolatos megnövekedett igényeket.

A C fájlorientált rendszer, egyaránt fájlként kezeli a billentyűzetet, a monitort vagy a lemezes állományokat. A C++ ezzel szemben objektumokat (osztályokat) rendel az említett eszközökhöz. Az I/O műveletekkel kapcsolatos objektumok az iostream könyvtárban kapnak elhelyezést, ennek használata megköveteli az iostrem.h fejlécállomány beszerkesztését.

A beviteli/kiviteli osztályok őstípusa az ios. Ennek leszármazottja az istream (input stream) — mely a beviteli műveleteket támogatja —, valamint az ostream (output stream); ez a kimeneti műveletekhez kötődik. Az istream és ostream osztálytípusok egyesítéséből származik az iostream osztály, mely kétirányú adatmozgást segít megvalósítani.

A C++ négy előre definiált stream objektummal rendelkezik:

— cin: istream típusú objektum, a szabványos beviteli eszközhöz (klaviatúra) kötődik. Megfelel a C-beli stdin-nek.

— cout: ostream típusú objektum, a szabványos kimeneti eszközhöz (monitor) van hozzárendelve. Megfelel a C-beli stdout-nak.

— cerr: ostream típusú objektum, szabványos hibakimenetet biztosít pufferelés nélkül. Megfelel a C-beli stderr-nek

— clog: ostream típusú objektum, szabványos hibakimenet-puffereléssel.

Tetszőleges adatszerkezet...

A hagyományos C magát az adatmozgást függvényhívások segítségével oldja meg. Például a printf( ) függvénnyel írhatunk a szabványos kimenetre, scanf( )-fel olvashatjuk a szabványos bemenetet. A C++ ezzel szemben operátorokkal; létezik beolvasó, illetőleg kiírató operátor. Az operátoros megoldás óriási előnye, hogy az operátorok átdefiniálhatók, így formálisan azonos módon oldhatjuk meg tetszőleges típusú (előre definiált vagy általunk létrehozott) adatszerkezet beolvasását és kiíratását is.

A kiírató operátor: <

Használata: ostream_objektum < adat

A kiírató operátor hatására az ostream objektumon keresztül (bal oldali operandus) az adat (jobb oldali operandus) a megfelelő fizikai eszközre továbbítódik. A kiírató operátor átdefiniált változattal bír minden beépített adattípus számára. Például:

```
int IntVal=1023;
double DoubVal=3.1415;
cout < IntVal;          // Képernyőn: 1023
cout < DoubVal;         // Képernyőn: 3.1415
cout < 100;             // Képernyőn: 100
cout < "Helló!";        // Képernyőn: Helló!
char Word[]="Ez egy mondat."
cout < Word;   // Képernyőn: Ez egy mondat.
```

A kiírató operátor ugyanolyan operátor, mint az összeadásé vagy a szorzásé. Hatását az operandusain fejti ki. Ennek megfelelően, ahogy van értéke az a+2 kifejezésnek, úgy pl. a cout<IntVal kiértékelése sem zárulhat eredmény nélkül.

A kiírató operátort tartalmazó kifejezés értéke az operátor bal oldali ostream operandusával egyenlő. Ily módon, mivel az operátor „visszatérési értéke" maga a kimeneti objektum,

a kiírató operátorok egymáshoz láncolhatóak. Megengedett tehát a következő kifejezés:

```
cout < "Az IntVal változó értéke: " < IntVal <
"\n";
```

Képernyőn: Az IntVal változó értéke: 1023

A kiírató operátorral memóriacímet is megjeleníthetünk. Például:

```
cout < "Az IntVal változó címe: " < &IntVal <
"\n";
```

Furcsa eredményt kapunk azonban, ha karaktermutató értékét próbáljuk megjeleníteni:

```
cout < Word; //A képernyőn: Ez egy mondat.
```

A jelenség magyarázata, hogy a char* pointert karaktersorozat kezdetének tekinti a rendszer. Ha csakugyan a mutató értékére vagyunk kíváncsiak, explicit típusmódosítást kell alkalmaznunk:

```
cout < (void*) Word;
```

A beolvasó operátor: >

Használata: istream_objektum > változónév

Hatására az istream objektumon keresztül, a megfelelő fizikai eszközről adatbeolvasás történik a megadott változóba. A beolvasó operátor átdefiniált változattal rendelkezik minden beépített adattípusra vonatkozóan. Például:

```
cin > IntVal; // Adatbeolvasás a
billentyűzetről az IntVal változóba.
cin > Word;// Karaktersorozat beolvasása a
billentyűzetről.
```

A kifejezés visszatérési értéke a bal oldali istream objektum, ily módon a beolvasó operátorok szintén egymáshoz láncolhatóak.

„Szabad" formák

A printf( ) és a scanf( ) függvények használata során megszokhattuk, hogy a kezelt adatok formátumát (mezőszélesség, tizedesjegyek száma, számrendszer) magunk választhatjuk meg. Nincs ez másként operátoros adatforgalom esetén sem.

Az aktuális adatformátumot a stream objektumok tárolják, jelzőbitek képében. A jelzőbiteket háromféleképpen állíthatjuk: néhány formátum beállítást tagfüggvényhívással is elvégezhetünk. Például a width( ) tagfüggvénnyel a mezőszélesség állítható. A flageket közvetlenül is módosíthatjuk a setf( ) és az unsetf( ) tagfüggvények alkalmazásával. Paraméterként a megfelelő bitmaszkot kell átadni, a szükséges részletek a kézikönyvekben megtalálhatóak. Általában azonban adatformátum-beállításra manipulátorokat használunk.

A C++-ban alkalmazható fontosabb manipulátorok

dec — Kiíratás/beolvasás decimálisan történik.
hex — Kiíratás/beolvasás (hexadecimálisan).
oct — Kiíratás/beolvasás (oktálisan).
flush — Üríti a streampuffert.
endl — Újsorjelet szúr az adatfolyamba, és üríti a puffert.
ends — Stringlezáró nullkaraktert szúr az adatfolyamba.
setfill(int c) — A mezőkitöltő karaktert c-re változtatja. (Alapértelmezés: space)
setw(int n) — A mezőszélességet n-re állítja.

Manipulátoron olyan eszközt értünk, mellyel egy stream objektum adatábrázolási módját változtathatjuk meg. Kifelé irányuló adatforgalom esetén a manipulátort a kiírató operátor felhasználásával „kiküldjük" az objektumnak. A manipulátor azonban nem jelenik meg a streamhez tartozó fizikai eszközön, csak a közvetítő objektumra van hatással. Beolvasás esetén a helyzet fordított. Ekkor a manipulátor „szűrőként" viselkedik a stream objektum és a befogadó változó között.

Manipulátorok használata esetén be kell szerkeszteni a programba az iomanip.h fejléc állományt is. Például:

```
int IntValt,FValt;
...
cout < dec < IntValt;// Az IntValt értéke
decimálisan jelenik meg
cout < hex < IntValt;// Az IntValt értéke
hexadecimálisan jelenik meg
```

A manipulátorok hatása a következő ellentétes értelmű manipulátor alkalmazásáig marad érvényben. Ily módon az előző utasítást követően

```
cout < FVAlt;
```

hatására FValt tartalma is hexadecimálisan íródik a monitorra. Az egyetlen kivétel ez alól a szabály alól a mezőszélesség állítás (setw), melyet minden alkalommal ismét használni kell. Például:

```
double DoubVal, dVal;
...
cout < setprecision(2) < setw(10) < DoubVal <
endl;
cout < setw(10) < dVal < endl;
```

A DoubVal és dVal változók értéke 10 karakterhelyen 2 tizedesjegy pontossággal kerül megjelenítésre. További példák manipulátorhasználatra a lemezmelléklet classio.cpp példaprogramjában találhatók.

„Kölcsönzött" értelmezéssel

Végezetül pedig az adatmozgató operátorok átdefiniálásának részleteiről kell még néhány szót ejtenünk.

Természetesen a kiírató és beolvasó operátorok is, a C++-beli operátorok többségéhez hasonlóan, igényeinknek megfelelően átdefiniálhatók. Az átdefiniálás következtében pl. a sorozat előző részében tárgyalt térvektor objektumot formálisan úgy olvashatjuk be/írathatjuk ki, ahogy azt egy double vagy int változóval tennénk. Például:

```
TerVekt v;
cin > v;
...
cout < v;
```

Az átdefiniálás konkrét lépései a lemezmellékleten található Vektor_3.cpp programban jól követhetők, itt csak néhány megjegyzésre szorítkozunk:

— Az operátorfüggvény első argumentuma a kiírató/beolvasó operátor bal oldali operandusa (a közvetítő stream objektum) kell, hogy legyen. Ennek következtében kiírató/beolvasó operátor nem lehet tagfüggvénye semmiféle általunk definiált osztálynak sem. Ha mégis szükséges, hogy az operátor hozzáférjen a kezelendő osztály protected mezőihez, akkor friend kapcsolatba kell hozni őket.

— Az operátorfüggvény második argumentuma a kiíratandó/beolvasandó adat.

— Átdefiniált operátorainkhoz saját manipulátorokat is szerkeszthetünk.

(A folytatásban a sorozat áprilisi része tárgyalja majd a C++ lemezes állománykezelő objektumainak használatát.)

Nagy Sándor

Visszafelé az időben

Elemek elemei

Mostani feladványunkban egy olyan írásrendszerrel ismerkedhetnek meg olvasóink, amely minden valószínűség szerint közvetlen elődje volt az ezerféle változatban élő indiai írásrendszernek.
Ami a lényeg: már ekkortájt is él a devanágaríból megismert rendszer, ha megjelenési formája nem is ugyanaz.
A köntöst cserélgették, foltozgatták — de ha valaki jobban odafigyel, még a devanágarí megújult köntösének mintázatában is észreveheti az ősibb motívumokat.

„Az Elemek Természetes Rendszere". Valamikor ilyen néven tanultuk Mengyelejev periódusos táblázatát. Korszakalkotó felfedezésének izgalmát nem mindenki képes újra átélni, ha tananyagként rögtön a kész rendszert tálalják elébe. Nekem szerencsém volt, mert a tudomány palotájának kapujába Sztrókay Kálmán kitűnő ismeretterjesztő könyvein keresztül jutottam el. Ő segített meglesni a gondolatok fogantatását.

Múlt havi feladatunk megfejtése nyomán talán akadt olvasóink között, akiben már megfogalmazódott a gondolat: milyen váratlan helyeken bukkannak elő szabályszerűségek a nyelvek világában. Hangok közötti relációkat, hangok struktúráját fedeztük fel, ha nem is mondtuk így ki. Talán közérthetőbb lesz, ha aránypárként is felírjuk egyik kimondatlanul maradt megfigyelésünket a hangok rejtett kapcsolatairól, ilyesféleképpen: t : n = p : m. Vagy egy másik relációról, amely múltkor szóba sem került: p : f = b : v. Ha következetesen végigvisszük ezt a szemléletet, a hangoknál is kisebb, elemibb egységeket állíthatunk elő, sőt le is választhatjuk őket konkrét hordozójukról.

Az elemek periódusos rendszere az elemek tulajdonságainak összehasonlításából kerekedett ki, és hová vezetett? Közös erőfeszítéssel eljutottak a kutatók (gondolati úton mindenekelőtt!) az atomok fizikai-kémiai világából egy szinttel mélyebbre, a részecskék birodalmába. A nyelvészet most vetette rá magát igazán a nyelvek struktúrájának, finomszerkezetének tanulmányozására. A hangtanban is, egyebütt is. Talán jut is valamire, ha számítógépesek és nyelvészek jobban megismerik egymás észjárását.

...És az indiaiak ezer évekkel a modern nyelvészek előtt milyen messzire jutottak már ezen az úton! Csoda-e, hogy az absztrakt gondolkodásra és a rendszeralkotásra egyaránt hajlamos indiaiak oly nagyra becsülték ezt az isteni szikrát?

Asóka király sziklába vésett feliratai

Viszonylag kései időkből maradtak ránk az első nyomok, valamikor Kr. e. 270 tájékáról. Amikor Asóka király az addig szokásos pálmalevelek helyett sziklákba vésette parancsait — úgy, ahogy buddhista meggyőződése diktálta, és négy nyelven, hogy mindenki megértse: az állatok feláldozása is tilos, az is gyilkosság. Fiát, Mahinda (Mahéndra) herceget a mesés csodavilágba, Lanka szigetére küldte, a messzire elhajózott szingalézek közé, hogy ők is fogadják el a buddhista tanításokat. A szingalézek akkor már évszázadok óta éltek távol régebbi hazájuktól, az Indus völgyétől, de nyelvüket mind a mai napig lényegében változatlanul megtartották.

A hazánk kétharmadát kitevő Srí Lanka (Ceylon) népsűrűsége — őserdei és kétezer méternél magasabb hegyei ellenére — kétszer akkora, mint Magyarországé. Buja növényzetére jellemzőek házmagasságú, 30-35 méterre felnövő óriási talipotpálmái, a folyópartokon tenyésző 25-30 méteres bambuszbokrai. A pálmák négy-öt méter széles leveleiből készítik azokat a keskeny lapokat, amelyekre ősi szokás szerint feljegyzik, majd idővel újra és újra átmásolják krónikáikat, szent irataikat. Ma India lakosságának kevesebb mint egy százaléka (!) buddhista, az egykor Indiához tartozó Srí Lanka viszont lényegében megmaradt a buddhizmus Rómájának.

Asóka feliratainak többsége hangtanilag jól kidolgozott brahmi írással készült, azzal, amit feladatunkban bemutatunk. Javasoljuk megfejtőinknek, hogy fokozatosan gyűlő ismereteiket foglalják mindjárt mátrixba — így jobban érzékelni fogják a rendszer kibontakozásának örömét.

A feladat

Négy csoportban adunk meg szavakat, három-háromféle módon: előszörre mai szingaléz nyelven, de latin betűs átírásban, másodszorra az ősi brahmi írással, végül magyar jelentésben. A szingaléz szavak kétféle írásmóddal megadott csoportjainál a csoportsorrend és az egyes csoportokon belüli sorrend más-más. A jelentések megadásának sorrendje viszont pontosan követi a brahmi csoportokét, csak technikai okokból nem állítottuk ezeket amazok mellé.

A latin betűs átírásban általában a szokásos betűmegfeleltetéseket alkalmaztuk (j = dzs, y = j, s = sz stb.), a magánhangzók hosszúságát azonban „magyar" ékezetekkel jelöltük. Egyetlen megjegyzés még: a nyelvheggyel ejtett T-családbeli családtagokat (t, th, d, dh, n) múltkori megállapodásunkkal összhangban 4-es számmal indexeljük, ha előfordulnak.

Kiejtés szerint:

1. csoport	2. csoport
a) budadina	a) baya
b) dinakara	b) dum
c) gurudina	c) dumkavuluva
d) karatáva	d) gajadata
e) kividina	e) gajakara
f) mahamuhuda	f) gajaya
g) maharaja	g) karaya
h) muhuda	h) kavuluva
i) rajanayá	i) usa
j) sanidina	j) usabaya

3. csoport	4. csoport
a) bekaya	a) dina
b) buda	b) kara
c) data	c) kavi
d) guru	d) kivi
e) kabala	e) maha
f) kana	f) nayá
g) kat4a	g) raja
h) kata	h) sani
i) katha	i) sáya
j) thalaya	j) táva

Magyar jelentések, a brahmi csoportok sorrendjében:

A) csoport	B) csoport
1. asszony; ragyogás; sugár	1. asszony
2. béka	2. csütörtök
3. beszélni	3. királyi kobra
4. fog	4. nagy király
5. fül	5. Nap
6. Jupiter; tanító	6. óceán
7. Merkúr	7. péntek
8. páncél; erő, energia	8. szerda
9. száj	9. szombat
10. talaj	10. tenger

C) csoport	D) csoport
1. csináló	1. ablak
2. éhség; este; kombiné	2. agyar
3. király	3. bika
4. kobra	4. elefánt
5. költői	5. füst
6. nagy	6. félelem; köcsög
7. nap (24 óra)	7. kémény
8. okosság	8. kéz, kar
9. Szaturnusz	9. magasság
10. Vénusz; költő	10. ormány

Megfejtéseiket, megjegyzéseiket lehetőleg még ebben a hónapban kérem a következő címre: Vargha Dénes, 1061 Budapest VI., Andrássy út 32.

„Azöreganyátokpihentagyú Kaleidoszkóp..."

Ezzel a megszólítással kezdődik Faragó Gergely érdekfeszítő levele, amely — hogy, hogy nem — csak másolatban jutott el hozzám, az év végi költözködés izgalmai után. Mivel azonban az indián/eszkimó írásrendszert januárban újra elővettük, bevárjuk még az erre reflektáló leveleket, és csak utána zárjuk le a témát.

Egy legény tehát mégis volt, aki már a szeptemberi feladvány alapján kihámozta a rendszert!

A devanágarí írások megfejtésére mások is vállalkoztak. Legérdekesebbnek Szatmári Éva levelét találtam, aki egyébként, úgy tűnik, ráérzett a Kaleidoszkóp-feladatok ízére. Most éppen nagy élvezettel fejtegeti a régebbi Alaplapok (érdekesnek és szórakoztatónak mondott) rejtvényeit. Az alábbiakban adjuk közre megfigyeléseit a hindi (általánosabban: devanágarí) írásrendszerről, zárójelben egyes helyeken kiegészítve, ill. pontosítva. [A szögletes zárójelbe tett szövegek a rovatvezető megjegyzései.]

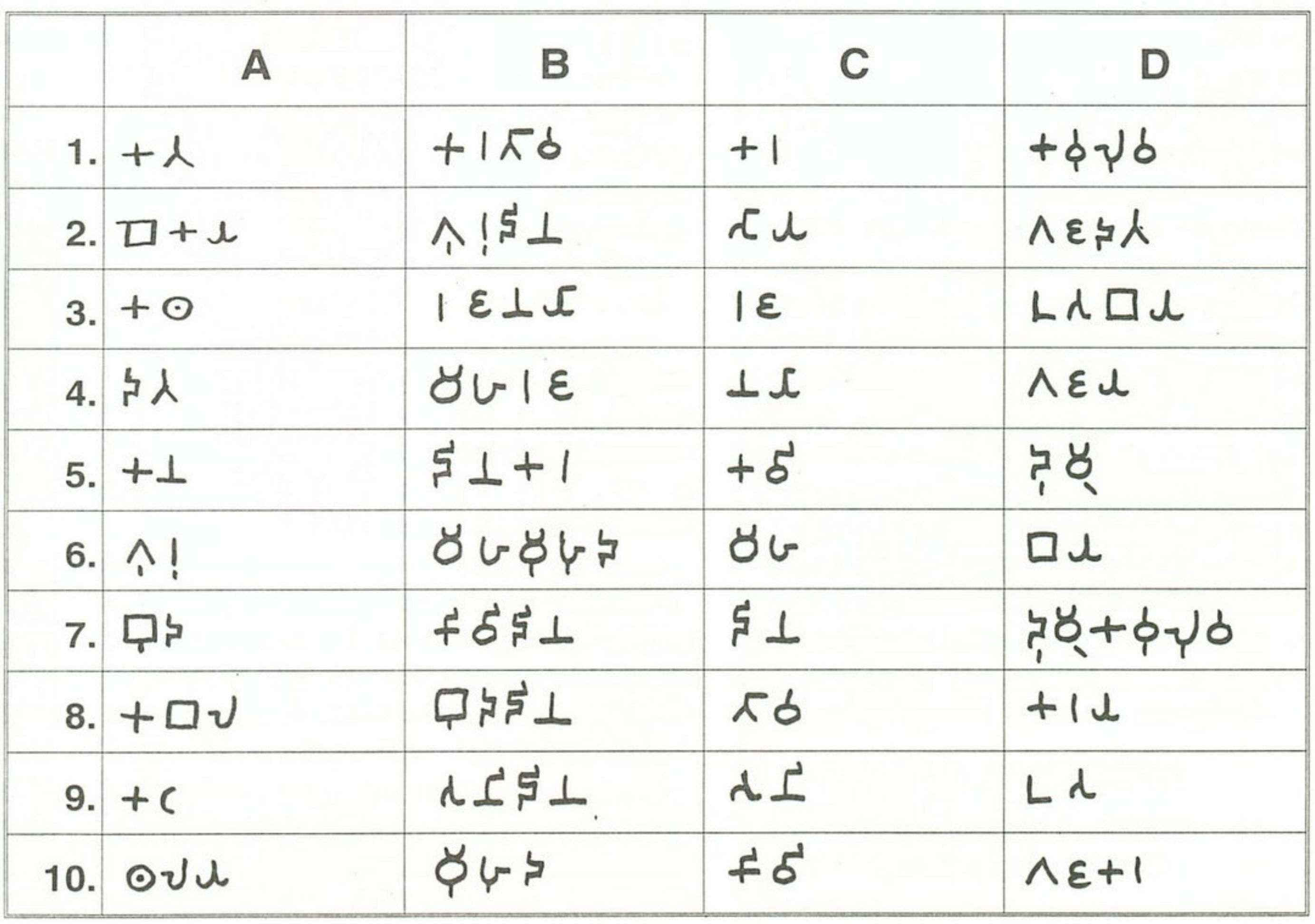
1. Ha a szó mássalhangzóval végződik, hangsúlyjel van alatta. [Ez a jel csak formailag olyan, mint a hangsúlyjel. Szanszkrit neve viráma, vagyis 'vég', 'megszakítás', 'lerövidítés'. Varga István keszthelyi olvasónk szíves közlése szerint magyar neve nyugvópont.] Ugyanez az eset, ha összetett szó első tagja végződik mássalhangzóval, pl. Egerszeg. Az „Alaplap" is összetett szó, a „p" betű alatt mégsincs vessző; talán azért, mert írásjele össze van vonva az „l" betű jelével. De akkor az Egerszeg szónál miért nincs összevonva az „r" és az „sz" betűk jele? [Megállapítása jogos. Az összevonásra nemcsak összetett szavaknál kerülhet sor, hanem mássalhangzó-torlódások esetén is, kötelező szabályok azonban nincsenek rá. A megszokottabb kapcsolódásokra alakultak ki ilyen összevont jelek. Az „r", mint decemberi példáinkban láttuk, kissé eltérő módon viselkedik.]

2. Az „o", „a" és „i" betűknek önálló írásjelük van még szó közben is. [Az „a"-nak nincs, csak a hosszú „á"-nak.] Az „i" betű esetén felcserélődik a sorrend, vagyis ha „i" következik egy mássalhangzó után, akkor előbb az „i" betűt írjuk le, és csak utána a mássalhangzót. [Megfigyelése tökéletes. Később adtunk példákat arra, hogy ez csak a rövid „i"-re igaz, a hosszú „í" jele nem előzi meg a mássalhangzó jelét. Hajjaj, ha tudná, hogy a mai szingaléz, malayalam, burmai és egyéb írásokban hogy jelölik az „e"-t és „o"-t!]

3. Ha a szó magánhangzóval kezdődik, akkor minden esetben kiírjuk az írásjelét. A szóvégi és szóközi „a" betűkre nincs külön írásjel. [A fogalmazás kissé pontatlan, de nehéz is precízen fogalmazni. Természetesen itt „a" hangokra gondol levélírónk.] Az „e" betűvel más a helyzet: szó elején természetesen kiírjuk a teljes írásjelet, szó közben azonban az előtte álló mássalhangzók tetejére egy vesszőt húzunk. A szóközi „u" betű jelölésére az előtte álló mássalhangzó alá egy farkincát húzunk.

4. Mássalhangzó-torlódásnál egybeírjuk a két [sőt több!] mássalhangzót úgy, hogy az elöl álló mássalhangzó írásjelének a végét (általában a függőleges vonalat) elhagyjuk, és hozzákapcsoljuk a második mássalhangzó teljes írásjelét. [Érdekes megfigyelni, hogy mi számít a mássalhangzók elmaradhatatlan elemének, különösen akkor, ha a brahmi jelekkel akarjuk egybevetni. Kiderül, hogy a függőleges elem csak „tartóoszlop", akárcsak a felül lévő vízszintes vonal. Sajnos, jó néhány mássalhangzó bemutatására nem is került sor, így olvasóink ezeknél nem végezhetik el önállóan az összehasonlítást. Különösen nagyfokú a hasonlóság például a „hegyesen ejtett" t hangok szótagjeleinek sorában.]

Varga István keszthelyi olvasónk egy profi versenyző biztonságával írja le az írásrendszerrel kapcsolatos tudnivalókat. Teljes magabiztossággal javítja ki a nyomdahibából eredő torzulásokat is. Megfejtése példaszerű, bár ismeretei láthatóan sokkal szélesebb körűek, mint



amiket tőlünk szerezhetett. Igen szépen írja a szókezdő „a"-k jeleit is, egy másféle betűtípus szokásai szerint. (A modern hindi szövegek nyomdajelei között rendszerint az ő „a" betűit találjuk.)

Faragó Gergelynek külön gratulálunk: ő eddig még minden feladatunkat megoldotta! A levelében említett „ÁÉ-rejtélyről" most csak annyit, hogy a magyar „á" helyett a szanszkrit hosszú „a"-t használtam, a magyar „é" és „ó" helyére azonban nem találtam megfelelő jelet. A szanszkrit nyelv „a" hangja ugyanis az ősi indoeurópai „a", „e" és „o" folytatása, ettől lett olyan monoton a-hangzású ez a nyelv. A szanszkrit „e" és „o" már másodlagosan származott az „ai" és „eu" hangkapcsolatok lerövidüléséből. Ezért viszont nincs hosszú párjuk, mert akkor visszaváltoznának „ai"-vá és „eu"-vá. Másik problémájáról: a „da" és „ra" szótagokhoz a következő függőleges nem tartozik hozzá, az csak az „á" hosszúságát jelöli. A „ga" és „sha" szótagoknál viszont a második elem is része az alapjelnek.

Vargha Dénes

A NEM KERESKEDELMI CÉLÚ EGYÉNI HIRDETÉSEK KÖZLÉSE INGYENES.

A kereskedelmi célú apróhirdetések tarifája gépelt soronként (azaz 60 karakterenként) 300 forint. A terjedelem alapján kiszámított összeget kérjük az Új Alaplap Kiadói Kft számlájára (Agrobank, 219-93789) átutalni, vagy postautalványon közvetlenül a kiadó címére küldeni (1538 Budapest, Pf. 571), és a hátoldalon feltüntetni, hogy „Új Alaplap, apróhirdetés". A befizetést igazoló szelvényt a hirdetési szöveggel együtt a szerkesztőséghez (a kiadóéval azonos címre) küldjék el.

A szerzői jogokat sértő szoftverhirdetéseket nem tesszük közzé. (Lásd erről bővebben 1994/januári számunkat.)

Eladó **C-64**: 1571 floppymeghajtó, 24 db programokkal teli és 16 db üres lemez, magnó, 9 db kazetta, 2 játékosnak való joystick, 2 cartridge, szakkönyvek, újságok. Tel.: 114-8511 (17-19 óra között).

Eladó **386-os IBM PC**: DX/40 MHz, 4 MB RAM, 1,2 MB és 1,44 MB FDD, 210 MB HDD, SVGA színes monitor, 101 gombos billentyűzet, egér, baby ház. Cím: Nyeső László, 9665 Vámoscsalád, Fő u. 51.

Eladó **ST-251** (42 MB-os) winchester vezérlőkártyával. Ára 6000 Ft. Cím: Gábor Zsolt, 1021 Budapest II., Hűvösvölgyi út 64. Tel.: 176-3333 / 199 mellék.

Objektumorientált programozás **Clipperben**: OBJECTS 2.0. Kérésre tájékoztatót küldök. Cím: Szűcs János, 4400 Nyíregyháza, Vasvári Pál u. 37. Tel.: (42) 313-568 vagy 312-222/1382-es mellék.

Eladók az **Alaplap 1990-1993 és a PC World 1992-1993** számai. Példányonkénti ára 150 Ft, egyben 100 Ft/példány az ára. Cím Nagy Péter, 1056 Budapest V., Molnár u. 20.

Adatmentés CD-re, streamerre, winchesterről, floppyról. Ugyanitt beszerzési tanácsadást, hálózattervezést és programkészítést is vállalok. Cím: Kovács Lajos, 1031 Budapest III., Vízimolnár u. 10. IV./33.

Számítástechnikai oktatás IBM PC gépen bármilyen témában. Beszerzési tanácsadást és programkészítést is vállalok! Fridl György. Tel.: 162-2070 (csütörtökön 16-18 óra között).

Stúdiómban megbízhatóan, ellenőrzötten lefordítom angol, német, francia és magyar nyelvről/nyelvre műszaki és közgazdasági folyóiratok cikkeit, hardver- és szoftverleírásait. Áfás számlát állítok ki. Cím: Szász György, 1035 Budapest III., Kórház u. 25. Tel.: 168-4874.

Keresek hardveres szakembert, aki beégetné egy Hercules videokártyába a CP 852-t. Tel.: 280-0833.

Keresem a Schneider LQ 3500 nyomtató leírását és gépkönyvét átnézésre, másolásra. Cím: Kern Miklós, 1037 Budapest III., Jablonka út 59./a.

Keresem az Alaplap 93/06 számát. Cím: Szabolcsi Mihály, 3950 Sárospatak, Kazinczy u. 1. I./22.

Eladók a COV c. újság 13-38-as számai , egy rossz állapotban levő camping bicikli (1000 Ft) és egy szintetizátor (2000 Ft) Cím: Kuliga Gábor, 5540 Szarvas, Szabadság u. 76. Tel.: (66) 311-761.

Keresem a **SimCity 2000** programot. Cím: Krausz Krisztián, 1045 Budapest IV., Zichy Mihály u. 3.

Eladó alig használt **Anitech videorecorder**. Esetleg elcserélném bármilyen számítástechnikai eszközre: monitorra, alaplapra, stb. Nagyobb értékű (legalább IBM AT 286) számítógép esetén kérem a video beszámítását. Ugyanitt keresek IBM AT 286-os gépet mono VGA monitorral és 40 MB-os winchesterrel. Cím: Monoki Zsolt, 5310 Kisújszállás, Kossuth Lajos út 9. IV./3.

Eladó reális áron egy keveset használt 14"-os színes **EGA monitor, vezérlőkártával.** Játékra kitűnő! Cím: Pápai Zoltán, 7257 Mosdós, Kossuth u. 9. Tel.: (74) 365-735 (7-15 óra között).

Keresek megvételre kézikönyvet vagy leírást a **Borland Reflex 2.0** programról. Vargha Dénes, 1061 Budapest VI., Andrássy út 32.

Geometria a matekórán

Absztrakt felfogás — gyakorlatias haszon

Előző számunk vezértémájának gondolatmenetét folytatják, illetve árnyalják a rovat mostani írásai.
A számítógépek gimnáziumi pályafutása érdekes ellentmondást mutat. Bár a számítástechnika tanítását gyakran matematikatanár vállalta fel, a számítógép a matematikaórán nemigen nyert polgárjogot.
Az alábbi cikk ennek fő okát is elmondja.

Az Új Alaplap olvasói bizonyára voltak már abban a helyzetben, hogy egy speciális feladat megoldásához megtanulták a legjobban hozzá illő nyelvet. Ebből kell kiindulni, ilyen problémákkal a matematikaóra bőségesen szolgál, tehát a programozás tanítását — ha a feladatot jól választjuk meg — bátran bevihetjük a matematikaórára. Mégpedig úgy, hogy eszközként nyúlunk a számítógéphez, akkor, amikor arra szükség van.

A gimnáziumokban a gép általában kizárólag a számítástechnikai órákon folyó, többnyire meddő programozási kísérletek szereplője maradt. A programozási feladat alapvető számítási algoritmusok megvalósítása, mint például a másodfokú egyenlet megoldása — emlékszünk még, hogy minden programozástechnikai tankönyv ennek folyamatábrájával kezdődött. Kérdés azonban, kellően érdekelt-e a tanuló abban, hogy egy idegen és elvont gondolkodásmódot elsajátítson, csak azért, hogy a másodfokú egyenlet megoldását vagy néhány szám átlagának kiszámítását most már számítógéppel is elvégezhesse, amikor ezekhez a műveletekhez bőven elég egy zsebszámológép. (Természetesen az ilyen típusú kiszámító funkció is hasznos lehet, amikor egy művelet sokszori ismétléséről, például egy sorozat „sokadik" elemeiről van szó.)

Ha tehát a siker reményében szeretnénk hozzáfogni a programozás tanításához:

— olyan legyen a probléma, hogy a tanulókat a gép használatára motiválja;

— olyan programozási nyelvet válasszunk, amely illik az adott problémához, és elsajátítása „megéri a fáradságot" az alapfeladat megoldása érdekében.

Relatív geometria

Szokásos geometriai szemléletünk egy alakzatot egészében vizsgál, vagy egy koordináta-rendszerben próbál elhelyezni, azaz például a kör általában „azon pontok mértani helye, melyek egy adott középponttól egyenlő távolságra vannak". Ezzel szemben Logo programozás közben magára a vonalra helyezkedve szemléljük az alakzatot; a kör így Logóban az a vonal, amely úgy jön létre, hogy 360-szor ismételve kicsit előre lépünk, majd egy fokot jobbra fordulunk, aminek az alábbi eljárás felel meg:

```
to kör
repeat 36! [fd 1 rt 1]
end
```

Nézzük most már meg, hogyan lehet a Logót becsempészni a matematikaórára. Vessük fel a következő (látszólag geometriai) feladatot. Osszuk egy kör kerületét öt egyenlő részre, a pontokat összekötve ötszöget kapunk. Ha pedig minden második pontot kötünk össze (ciklikusan, amíg lehet), szabályos csillagötszöghöz jutunk.

Ugyanezt megismételve (hat, nyolc, kilenc stb. ponttal) és különböző hosszúságú lépésekkel, érdekes tapasztalatokhoz juthatunk.

Hogyan függ tehát a kapott ábra a pontok számától (n-től) és a lépés hosszától (k-tól)? Papírral és ceruzával a tanulók nem tudnak elég sok ábrát rajzolni ahhoz, hogy a tökéletes választ megsejtsék — és itt hívjuk segítségül a Logót. A csillag problémája elég érdekes ahhoz, hogy az ábrák megrajzolásához megtanulják a Logo programozás alapjait. A kész eljárás mindössze ennyi:

```
to csillag n k
  repeat n [fd 50 rt *k *
360) ^ n]
end
```

Az eljárás most már neve és paraméterei megadásával — akár egy parancs — hívható, azaz a bevezető ábrát csillag 5 2 rajzolja.

Az eljárás rövidségére ismét a relatív geometriai szemlélet ad magyarázatot: n csúcs esetén n-szer kell ismételni az előrelépést, egy elfordulás nagysága pedig az összes fordulat, azaz k*360 n-ed része lesz.

Másképpen, más megvilágításban

Ha a Logo programozás elemeivel már megismerkedtünk, egyes geometriai fogalmak — ilyen például a körhöz húzható érintő — jobb megértéséhez hívjuk segítségül a Logo szemléletmódját. Az érintőt általában a szelő valamiféle határhelyzeteként fogjuk fel. A Logóval bemutathatjuk, hogy az érintő milyen közvetlen módon jellemzi a görbét. Erre a felismerésre különben a mozgások leírásánál elengedhetetlen szükség van.

Arról az egyszerű jelenségről van szó, amikor például egy korcsolyázó az ugrás után „jeget ér", a forgás során leírt pálya érintője mentén csúszik tovább, azaz a görbe pályán mozgó test aktuális iránya éppen az érintőé. Mivel a logós rajzolás során rajta vagyunk magán a görbén, érintőt egész egyszerűen úgy kapunk, hogy valamely pillanatban az éppen adott irányban egy egyenest húzunk. Az eredeti körrajzoló eljárást egy billentyűfigyelő rutinnal kiegészítve elérhetjük, hogy a kör rajzolása során billentyűnyomásra az adott helyen egy érintőt húzhassunk. Hogy a beavatkozásra legyen elég időnk, használjuk a slowturtle utasítást, vagy más lassító eljárást! Az érintőhúzásra külön eljárást írunk, így ez már modulokból építkező program lesz:

```
to érintő
  pu bk 4! pd fd 8! bk 4!
end
```

```
to kör
  slowturtle
  repeat 36![if
key?[érintő make "a
readchar] fd 1 rt 1]
 fastturtle
end
```

Mindezt kiegészíthetjük a hagyományos érintőfelfogást bemutató programmal, amikor először egy kört rajzolunk, majd a kerület egy pontjához húzott sugárra merőlegest állítunk:

```
to kör1 r
  pu fd r rt 9! pd
  repeat 36![fd *r * 3.14(
^ 18! rt 1]
  pu lt 9! bk r pd
end
```

Most készen áll a kör, középen álló teknőccel. Az eljárás bonyolult, mert ebben az esetben nem a körvonalhoz, hanem a középponthoz képest kell tájékozódnunk. Figyeljük meg az eljárás első és utolsó sorát, amelyekben két, egymással „tükrös" utasítássort találhatunk. Ezzel érjük el az ún. állapot-átlátszóságot, azaz hogy a teknőc az eljárás előtt és után azonos helyzetben legyen. Ha most érintőt akarunk húzni, valamilyen irányban ki kell mennünk a körvonalra (sugár), ott 90-et fordulni, és ekkor már meghívhatjuk az érintő eljárást. Látható, hogy ez mennyivel bonyolultabb az első, kör nevű rutinnál — az is igaz viszont, hogy ebben az esetben ismerni fogjuk az érintő pontos helyét.

A fenti példa tehát azt mutatta, hogy az új szemlélettel egy már ismert fogalom egészen más megvilágításba kerülhet.

Szemléletünk erősítésére

Az újabb Logo-változatokban lehetőség van több teknőc egyidejű mozgatására, így szimmetrikus ábrákat készíthetünk. Tengelyesen szimmetrikus rajzokat kapunk, ha két párhuzamos alaphelyzetű teknőcöt mindig azonos irányban, de ellentétes forgással mozgatunk, míg középpontosan lesz szimmetrikus az eredmény, ha ellentétes irányból induló teknőcökkel azonos utasítássort hajtunk végre.

Ez a két fenti transzformáció körüljárásváltó, illetve körüljárástartó tulajdonságának Logo-megfelelője. Az eljárások:

```
to tengelyes
  tell [0 1] st
  repeat 3[tell[0 1] fd 50
tell 0 lt 120 tell 1 rt
120]
end

to kp/os
  tell [0 1] st tell 1 rt
180
  tell [0 1] repeat 3[fd
50 rt 120]
end
```

A számítógéppel való találkozás az iskolában tehát nem szükségszerűen öncél. Az itt vázolt módon a tanulók belekóstolhatnak az alapfokú programozásba.

Ugyanakkor a gimnáziumok mai gyakorlatában szerepel a felhasználói ismereteket nyújtó képzés is. Ha ezeken az órákon is gyakorlatiasságra törekszünk, méltán remélhetjük, hogy az iskolát a számítógéppel alkotó kapcsolatban álló diákok hagyják el.

Földes Petra

Modemen keresztül tanulni

Statisztikák szerint a magyar informatikai szakemberek több mint 75%-a tanult a Számalk diplomát adó kurzusain, felsőfokú és középfokú képesítést adó tanfolyamain, „state of art" szemináriumain és egyéb rövid kurzusain. A 18-30 éves korosztályból évente közel 6000-en választják a Számalk különböző szintű képzési lehetőségeinek valamelyikét. Úgy gondoljuk, hogy — elsősorban vidéki olvasóink — érdeklődésére tarthat számot a távoktatáson alapuló Gábor Dénes Műszaki Informatikai Főiskola.

A második éve működő magán távoktatási főiskola egyre népszerűbb a számítástechnika iránt érdeklődők körében. Az átjárható, „post secunderi" képzést megvalósító főiskolán közel 1500-an tanulnak. A hallgatók közel 60%-a magánemberként iratkozik be a főiskolára, hisz elmúltak azok az idők, amikor a vállalatok csakúgy „nyomták" a hallgatókat. Természetesen kell tandíjat is fizetni, kb. 80 000 forintba kerül egy év a Gábor Dénes főiskolán. Lehet „részletre" is tanulni, de újításként — s főleg a Számalkon belüli adminisztráció megkönnyítésére — hamarosan bevezetik hogy valamelyik bank személyi áruhitelt (!) ad azoknak a hallgatóknak, akik nem tudják egy összegben kifizetni a tandíját. További könnyítést jelent, hogy az oktatási költség — pontosabban a hivatalos felsőoktatási havi tandíj maximumával azonos összeg — leírható az adóalapból.

A fizetési könnyítések és adóalapcsökkentő lehetőségek ellenére nem csekély az okulásért fizetendő összeg. Azonban meglehetősen komplex képzést kapnak a pénzükért a hallgatók: az alapvető informatikai ismeretek mellett gazdasági (üzleti) és menedzseri (project management, döntéstámogatás) képzésben is részesülnek.

Tanítási módszereiben is sok újat hozott a főiskola. Tekintettel arra, hogy — Magyarországon elsőként — távoktatáson alapul az intézmény, így az oktatásban nagy szerepet kap a modem és a terminál. Minden hallgató, akinek van telefonja — 25 000 forint ellenében — kibérelheti ezeket az eszközöket (a bérleti díj lejön az éves tandíjból). Így a hallgatók otthonukból is elelsajátítják a tananyagot: lakásukból, a nap bármely szakaszában rákapcsolódhatnak a Gábor Dénes főiskola 14 vidéki hálózatára. 4000 videokazettából választhatják ki az őket érdeklő témákat. A kazettákon előadások szerepelnek a legkülönbözőbb témakörökből, így például 16 órás a matematikai videoanyag. A kazetták anyaga szabadon másolható, így akár többször is vissza lehet nézni egy előadás homályos pontjait vagy akár „háttérvideózás" keretében lehet memorizálni az ismereteket. De természetesen lehet konzultálni is a távoli oktatókkal — telefonon és faxon.

Úgy érezzük, hogy a Gábor Dénes Főiskola olyan új lehetőséget nyit meg a felsőfokú számítástechnikai képzésben, amely mindenképpen sikerre számíthat, különösen a számítástechnika fellegváraitól távol eső vidéki kisvárosokban és településeken. Sőt, a főiskola a munkaerő-átképzés autodidakta formáját is megtestesíti.

Tudják ezt a munkaerő-átképző központok is, s ha valaki elég kitartóan kilincsel náluk, akkor elvileg rábírhatók, hogy az átképzés költségeit vállalják magukra.

Sziebig Andrea

SZÁMÍTÁSTECHNIKA KULCSRAKÉSZEN!

- 386-os, 486-os, 586-os számítógépek minden kiépítésben. (3 év garanciával!)
- Notebook, laptop gépek.
- EPSON, Hewlett-Packard nyomtatók.
- DISCOVERY, MICROCOM és US ROBOTICS modemek.
- Szoftverek széles választéka.
- NOVELL hálózati szoftverek, hálózatépítés.
- Tartozékok, kiegészítők, szakkönyvek széles választéka.

QWERTY

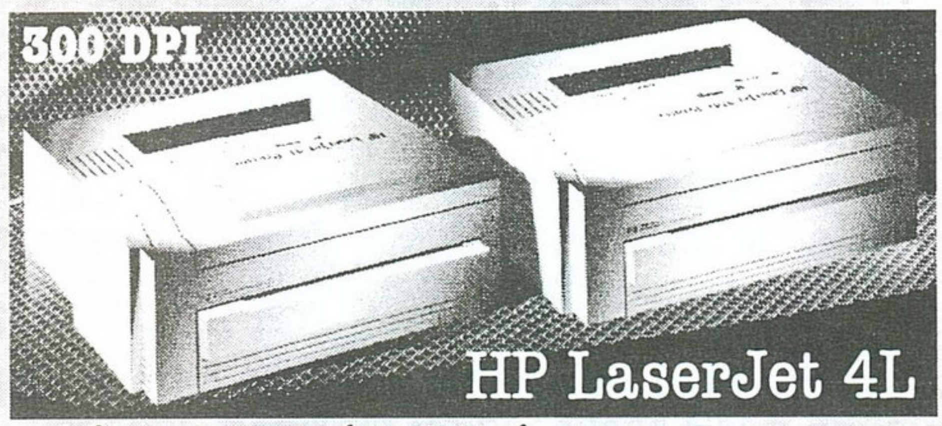
High Tech Kft.
1114 Budapest, Bartók Béla út 9.
Telefon: 186-8858, 185-2687, 186-9285
Fax: 185-2687
BBS: 266-2292 BUDAPEST BBS (10 vonal)

NE FELEDJE: nevünk ott található az Ön számítógépének billentyűzetén is!

INFORMÁCIÓKÉRÉS: A0336 ▼

HEWLETT® PACKARD

SZAKÁRUHÁZ



ÚJ SZOLGÁLTATÁSAINK:

ÚJ!!!
- ingyenes, 48 órán belüli házhozszállítás (*)
- helyszíni üzembehelyezés
- magyarnyelvű használati útmutató (**)
- helyszíni installálás (***)

* 40.000 Ft. felett, Budapesten
** egyes tipusokhoz (pl.: DJ 510, 4L,..)
*** fizető szolgáltatás

Nyitvatartás:
H-P: 08-18 óráig

VECTRA
HEWLETT PACKARD
SZAKÁRUHÁZ
1091 Budapest, Üllői út 5.
Tel.: 218-8800 Fax.: 218-8801

INFORMÁCIÓKÉRÉS: A0341 ▼

A SuperNOVA

ELSŐ HAZAI SZOFTVERHÁZA

A szupernóva — fény és energia.
Ezt az új energiát Ön is hasznosíthatja.
A SuperNOVA kínálta nagy hatékonyságú, objektumorientált, grafikus fejlesztőkörnyezet, a 4GL nyelv, a CASE-kapcsolat, az adatbázisfüggetlenség ideális lehetőségeket nyújt az alkalmazásfejlesztés minden területén.
A feladat kijelölésétől a rendszerterven át a platformfüggetlen alkalmazás elkészültéig igyekszünk méltóak lenni a szupernóvák sebességéhez.

MemoLuX®
1443 Budapest, Pf. 228
Telefon: 183-2935, 183-3111 Telefax: 163-5079

Felelős ügyvezető: Ivanyos János

Fejlesztői szolgáltatásunkat igénybe vevő partnereink részére ingyenes szakmai továbbképzést tartunk Hollandiában.

INFORMÁCIÓKÉRÉS: A0329 ▼

Nem csak a hardvereseké a világ...

Az újdonságok között most a szoftveres beszámolók kerültek túlsúlyba. Az irodai adminisztrációt könnyítő DA-Dosszié mellett CD-n forgalmazott két szoftvert (Jogtár, Telefonkönyv) vallattunk, megismerkedünk egy sikeres hazai fejlesztésű banki szoftverrel (IBA), majd bemutatjuk egy nagy reményekkel induló alkalmazásfejlesztő eszköz (SuperNOVA) főbb jellemzőit. Nem feledkez(het)ünk meg azonban a hardverről sem, szó lesz a Miro és a Sony legújabb termékeiről, továbbá a HP szokásos év eleji nagyszabású termékbejelentéseiről.

Jogszabályok, telefonkönyvek — tenyérnyi helyen

Az állandóan változó jogszabályok, és az országszerte bővülő telefonhálózatok újfajta számítástechnikai szolgáltatásokat hívtak életre. Több mint egy évvel ezelőtt a Kerszöv Kft gondozásában került piacra CD-lemezen az éppen hatályos jogszabályok számítógépes információs rendszere (ComPLEX) — erről az Alaplap júniusi számában, a Szoftvertéka rovatban olvashattak értékelést az érdeklődők. Ősztől pedig a Tele-Data Kft rukkolt ki egy újfajta CD-szolgáltatással, az országos, hivatalos telefonkönyv adatait tartalmazó adatbázissal.

Változó világunkban azonban mindkét CD alapos frissítést igényel, így nem árt megismételnünk: a jogtár havonta újul meg, minden hónap 20-áig egy új CD-t kapnak a felhasználók az aktualizált jogszabályokról. A még csak félig telt CD tartalmazza a hatályos magyar joganyag teljes szövegét: mintegy 60 darab 600 oldalas könyvet. A négyféle, átjárható adatbázisba (Hatósági Jogszabályok, Bírósági Döntvénytár, Magyar Közlöny, Törvények Indoklása) szervezett jogtár állandóan bővül új szolgáltatásokkal is. Például külön adatbázisként teszik közzé az Országgyűlés által elfogadott törvények indoklását, egységes szerkezetben szerepelnek a Legfelsőbb Bíróság valamennyi — polgári, gazdasági, munkaügyi, közigazgatási, büntető — kollégiumának hatályos irányelvei, és az önkormányzati értekezletek állásfoglalása.

A jelenleg 95 kibocsátótól 21 ezernél több joganyagot tartalmazó tár 3000 forintos havi frissítés ellenében valamennyi minisztériumban, polgármesteri hivatalban napi munkaeszköz, és a budapesti APEH összes termináljáról is elérhető az aktualizált, hatalmas joganyag. Eddig Székesfehérvárott gyártották a CD-ket, sajnos azonban nem minden típusú CD-ROM olvasta el azokat. A kellemetlen helyzet kiküszöbölésére a Kerszöv gyártót váltott, s immár a Sony salzburgi gyára állítja elő a CD-ket, ráadásul 10-15%-kal olcsóbban, mint idehaza.

A telefonkönyves CD-s szolgáltatást is beharangoztuk már az Alaplap októberi számában, de a CD-s telefonkönyvet csak most lapozhattuk fel, s igen jót játszottunk tesztelése közben. Keresgettünk ismerőseink neve alapján, lakcíme és irányítószáma szerint, szakmájuk alapján. Megnéztük, hogy telefonszámuk ismeretében mi mindent ad ki róluk az adatbázis. Természetesen ezeknek a szempontoknak a töredéke vagy kombinációja alapján is megkaptuk a keresett adatokat. A mintegy másfél millió adatot tartalmazó adatbázisban alig van hiba, persze nekünk „sikerült" ilyet is találnunk. A Tele-Data természetesen csak azokat az adatokat viheti be, amelyekett megkap a Matávtól. Ha az előfizető nem jelenti be például, hogy elköltözött, vagy a Matáv belső adminisztrációjában elkallódik az irat, akkor bizony a monitoron sem jelenik meg a helyes adat.

A Telefonkönyv sajnos tudatosan beépített korlátokat is tartalmaz. Lehetővé teszi ugyan a nyomtatást és a kikeresett adatok elmentését fájlba, de csak képernyőnyi terjedelemben. Ezzel talán azt akarják megakadályozni, hogy az adatbázist más cégek „üzletszerűen" hasznosítsák, pedig így csupán a saját kapcsolatrendszerük adatainak frissen tartására törekvő cégeket fosztják meg a kényelmes hozzáférés lehetőségétől. A kényelmetlen hozzáférés lehetősége mindig is megvolt, a telefonkönyv nyilvános adatbázis, régebben néhány gépírónőt ráállítva, manapság a Recognita szövegolvasóval beszkennelve bárki kigyűjtheti belőle a neki szükséges címeket és telefonszámokat.



LION ELECTRONIC

... emberbarát elektronika

1036 Budapest III., Tanuló u. 1. • Telefon/Telefax: 188-3222, 168-6239



Nyomtatók

CANON

CANON BJ-10SX fehér/fekete	29 900 forint
CANON BJ-200	39 900 forint
CANONBJ-300 A/4 (opció: AUTOCAD Driver)	61 800 forint
CANON BJ-330 A/3 (opció: AUTOCAD Driver)	75 900 forint
CANONBJC-600 Color	88 900 forint

HEWLETT-PACKARD

HP 510	34 900 forint
HP 550 Color	67 800 forint
HP 4	189 900 forint
HP 510 dupla élettartamú fekete patron	3 200 forint
HP 550 színes patron	3 500 forint



Áraink az áfát nem tartalmazzák!
Az árváltoztatás jogát fenntartjuk, deviza-árfolyamváltozás esetére!

INFORMÁCIÓKÉRÉS: A0325 ▶

MEGBÍZHATÓSÁG, ÜZEMBIZTONSÁG, sokoldalú SZERVIZ

!ÚJDONSÁG!

Rejtett audio/video megfigyelő-, felügyelő- és ilyent detektáló rendszerek
Helyiség-, telefonlehallgatás elleni védelem.

Viszonteladókat várunk.

Koaxiális- és üvegszálas hálózatok, adatmentő és -védelmi rendszerek
üzenetrögzítős fax/modem kártyák, multimédia audio/video rendszerek

TCC COMPUTER WIEN BUDAPEST

Számítógépek, hálózatok, szerverek
INTEL, NOVELL, Microsoft
3COM, OPTICOM, JET PROPULSION
HP, STER nyomtatók
VASCON biztonsági rendszerek
AITECH audio/video

1117 Budafoki út 70.

Tel: 166-7698, 166-7044 Fax: 166-7698

INFORMÁCIÓKÉRÉS: A0310 ▶

ARTEC

MOUSE, KÉZI SCANNER

A legnagyobb hazai választék.

A mouse-okra örök garancia!

VETÉLYTÁRS NÉLKÜL

PINNACLE MICRO
THE OPTICAL STORAGE COMPANY™

TAHOE-130™ hordozható magneto-optikai meghajtó
PMO-130™, PMO-650™, SIERRA-1.3GB™ magneto-optikai meghajtók
JUKEBOX-ok 93 GB-ig
RCD-202™ írható CD-ROM meghajtó
CD-ROM meghajtók **(DS, KPh)**

Magneto-optikai lemezek
Üres CD (CD-R) lemezek

IBM PC SUN MAC DEC

AVISION

PROFESSZIONÁLIS A4-ES SCANNEREK:

képfeldolgozás, dokumentum archiválás, karakterfelismerés, faxmunkaállomás.

AV100 roll scanner (600dpi, f/f)
AV660 (1200dpi/24bit szín)
AV680 (1600dpi/24bit szín)
AV680G (1600dpi, f/f)
AV800 síkágy/gyors lapadagoló (1200dpi/24bit szín)

Minden készüléket képfeldolgozó programmal szállítunk.

Rendelhető: dia feltét, OCR program

ZyXEL

General DataComm

alapsávi/bérelt/kapcsolt modemek, faxmodemek, RS-422/485 átvitel, adatátviteli és faxprogramok, LAN faxrendszerek, távvezérlő és felügyelet nélküli adatlekérdező rendszerek, hívás-szétválogatók, auto-on-box, programozható időzítők

MODEM IDŐK

BEST

HUMANSOFT

HUMANsoft Elektronikai Kft.
1149 Bp. Angol u. 24/b.
Tel: *163-2879 Fax: 251-3673

INFORMÁCIÓKÉRÉS: A0320 ▶

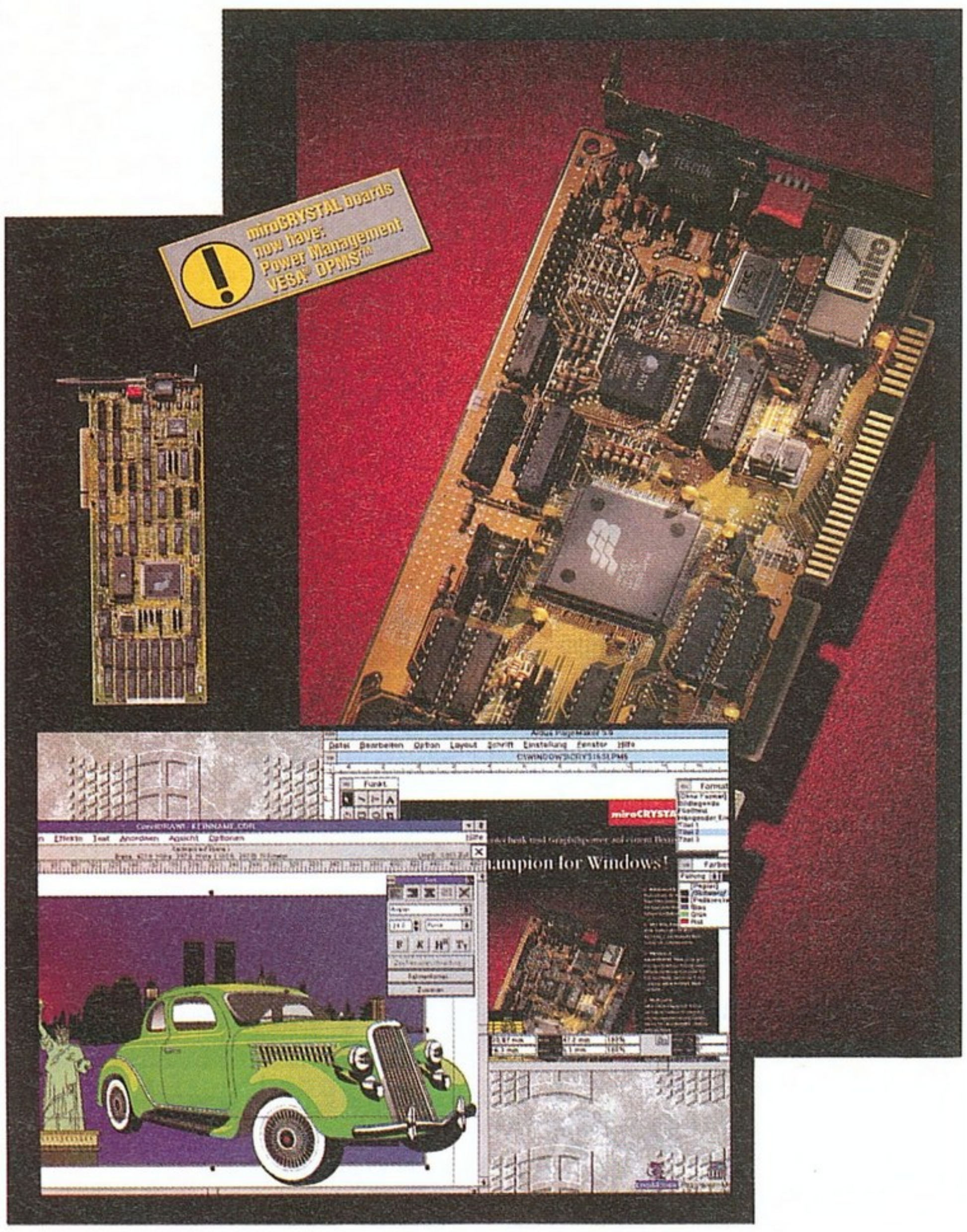
Másik furcsaság, hogy ezt a Telefonkönyvet nem lehet megvenni, csak bérelni az adatbázis-karbantartó Tele-Datától (az adatbázis a Matáv tulajdona). Az 5000 forintos bérleti díj első ránézésre méltányosnak látszik, de mert azt negyedévente kell kifizetni, az évi 20 000 Ft kicsit soknak tűnik akkor, ha az adatbázist csak arra használhatjuk, hogy időnként „fellapozunk" benne egy-egy telefonszámot. (Másként pedig ugyebár nem szabad!?) A bérleti idő letelte után pedig (elvben) hozzá sem lehet férni az adatbázishoz...

A felhasználó jogosan hiányolja a rádiótelefonok visszakereshetőségét is. A Westel határozott kívánságára a 60-as körzetszámú lista egyelőre tabu, így az a jegyzék, amely a szakmai, üzleti élet infrastruktúrájához való kacsolódást tenné lehetővé, továbbra is titokban marad, s nem egészen érthető, hogy milyen alapon, hiszen ezek a telefonszámok sem titkosak.

Miróval és Sonyval megerősödve

Akik régebben vásárolták meg személyi számítógépüket, szomorúan tapasztalják, hogy a 286-os gép és a 386-osok egy része nem nagyon alkalmas a Windows-alapú programok futtatására. Ezért örültünk annak, hogy a hazai piacon is megjelentek az alacsony árfekvésű Miro grafikus kártyák, amelyekkel a régi PC-k még „tuningolhatók".

A Miro disztribúciós feladatait magára vállaló Számalk Hardware Disztribútor a grafikus kártyákat nemcsak Windows-gyorsítóként forgalmazza, hanem DTP és CAD/CAM alkalmazásokhoz is ajánlja. A miroCRYSTAL kártyacsalád (8S, 16S, 24S, 32S) valamennyi tagja készül mikrocsatornás kivitelben, és a lokálbuszos (VL, PCI) vagy ISA/EISA architektúrán alapuló grafikus kártyák Macintosh gépekbe is behelyezhetők. A miroCRYSTAL legkisebb (8S), legolcsóbb (30 000 forint körüli) tagját elsősorban gyorsítókártyának ajánlják a forgalmazók. Valószínűleg ebből a típusból adnak el majd a legtöbbet. A család eggyel nagyobb tudású tagja (16S) a grafikus és CAD alkalmazások (CorelDRAW, AutoCAD, FreeLance, stb.) kiegészítője. A 24S TrueColor minőséget biztosít AutoCAD és 3D-Studio típusú szoftverekhez, a 32S pedig már egy „mindentudó" eszköz.



A grafikus kártyák mellett új Miro monitorok is megjelentek a disztribútor kínálatában. Az igényes, nagyfelbontású (1280x1024), 15"-os és 17"-os MAG monitorok kedvező árfekvésűek (a 15"-os megközelítőeg 70 000 Ft, a 17"-os trinitronos 139 000 Ft), jó Windows-környezetet biztosítanak, s akár helyettesíthetik is a Compaq gépek képernyőit. A high-end alkalmazásokhoz (DTP, CAD/CAM) is kínál a Miro monitort, a Proofscreen márkanév alatt készülő, 20"-os és 21"-os Sony OEM monitorokat.

A Sony átszervezte magyarországi jelenlétét. Eddig csak az Albacomp és az XL állt kapcsolatban a Sony Európával, idén azonban már a Számalk Hardware Disztribútor képviseli termékeit. Négyféle CD-meghajtót értékesítenek (külső és belső AT buszos, SCSI); a 128 és 650 Mbájtos, SCSI felületű magneto-optikai (MO) változatokat elsősorban archiválási célokra ajánlják.

Ez utóbbiak hiánypótló szerepet töltenek be a — Compaq gépeket is képviselő — disztribútor kínálatában, mert a Compaq nem gyárt MO meghajtókat, s a 650-es jól igazodik a Compaq-szerver konfigurálásához.

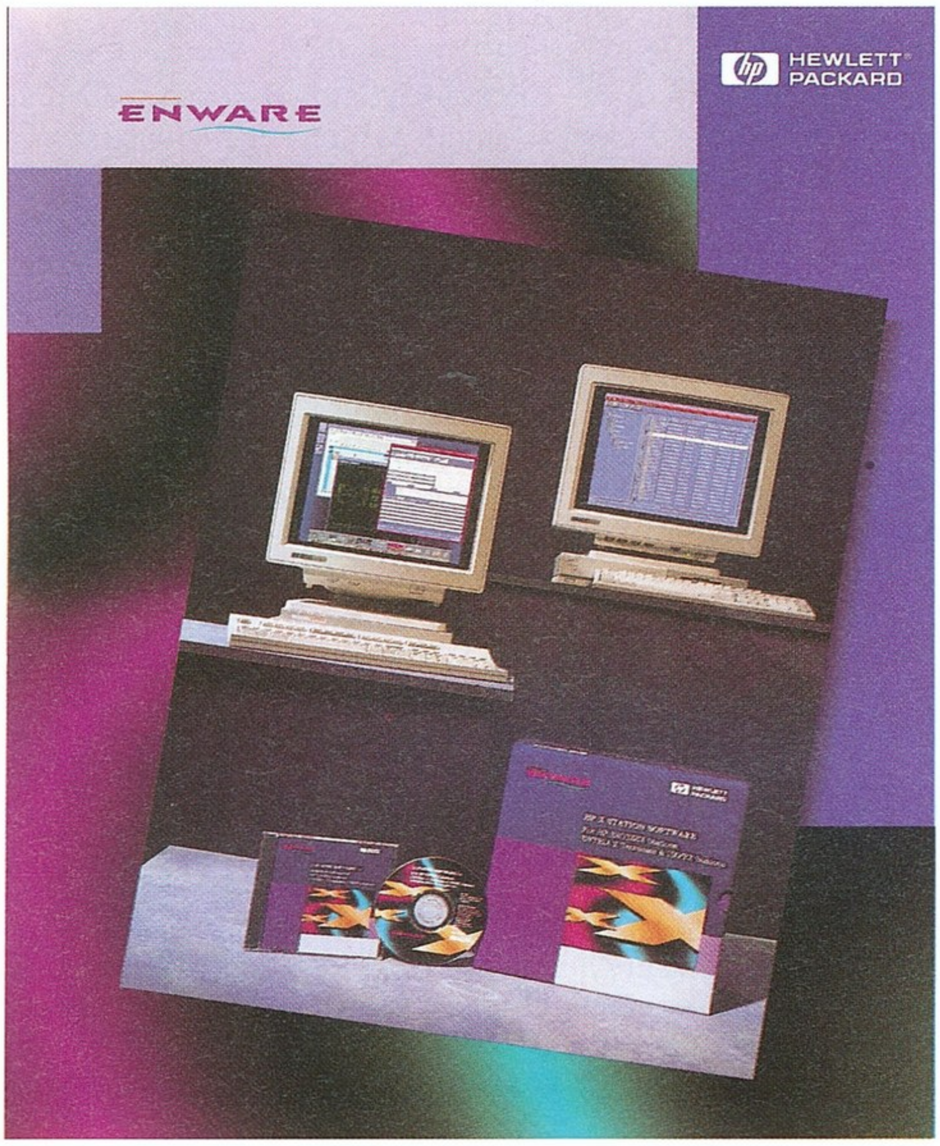
A multimédia térhódításával, és elsősorban a Microsoft, Novell, SCO szoftverei méretének növekedésével az új adathordozók iránt igencsak megnőtt a kereslet. Szakemberek szerint valóságos CD-felfutás várható Magyarországon is.

Bankvilág — Magyarországon

A számítástechnikai beszerzéseknek a lehetőleg „hazait" támogató gyakorlatától eltérően pénzintézeteink előszeretettel választják a nagy referenciákkal rendelkező külföldi informatikai rendszereket a magyar fejlesztésű banki alkalmazásokkal szemben. (Hasonló tendencia figyelhető meg — elsősorban a vegyesvállalatok részéről — a gazdasági szoftvereket illetően.) A harmadik generációs technológiával készült nyugati banki szoftverek a lassan változó nyugat-európai igényeket igaz, hogy kitűnően kielégítik, azonban a rendkívül gyorsan változó kelet-európai gazdasági környezettel nehezen tartanak lépést.

Erre a problémára több honi — szoftverfejlesztéssel foglalkozó — cég is ráérzett, s felismerték, hogy a magyarországi pénzintézeteknek olyan korszerű információs rendszereket kell kidolgozniuk, amelyek negyedik generációs szoftverfejlesztési technológián alapulnak, a mi gazdasági és törvényi feltételeinknek megfelelnek, adaptációt nem igényelnek, és gyorsan változtathatók.

Az integrált banki alkalmazások közül talán csak egyetlen komplex rendszer (az IBA) veszi fel a versenyt a külföldi szoftverekkel. Az értékpapír (Broker-Int), üzletági (Merk-Int), számlavezetési (IBA-Bag), kockázatkezelési (Risk-Int) és információs („I" Project) rendszerekből álló IBA jelentős referenciákkal is büszkélkedhet. Ez komoly fegyvertény a szoftverfejlesztő Integra kezében, mert legtöbb bankunk



elveti a „magyar áru" alkalmazásának gondolatát mindaddig, amíg annak nincsenek honi banki „ajánlólevelei".

A 22-es csapdájából többszörösen sikerült kitörniük az Integra szoftvereinek, hisz az IBA-Bag integrált számlavezetési rendszert használja a Magyar Külkereskedelmi Bank és az Investbank. A Kereskedelmi és Hitelbankban pedig Magyarország legnagyobb online értékpapírrendszere üzemel: a Broker-Int unixos változata a K&H 80 fiókjában, X.25 vonalon keresztül működik.

A moduláris felépítésű rendszer lefedi az értékpapírokkal kapcsolatos valamennyi tevékenységet. Jelenleg az értékpapírpiacon létező bármilyen értékpapír kezelésére alkalmas a rendszer, függetlenül az értékpapír megjelenési formájától (fizikailag kinyomtatott, számlán nyilvántartott értékpapír). A moduláris felépítés (jegyzés, részvénykönyv, megbízás, piacvezetés, esedékességek, értékességek, értékpapír-átváltás, ügyféláthelyezés, pénztár, értéktár, értékpapír-letiltások, likvid, befektetési alapkezelés) okán a rendszer elemei használhatók önállóan, és tetszés szerint egymáshoz illesztve is. Mivel az egyes értékpapírügyletekből adódó tételek kontírozhatók, így adódik a kapcsolat a főkönyvi könyvelési rendszerrel.

Akad tehát Magyarországon olyan szoftveres fejlesztőgárda, amely nem érezte reménytelen vállalkozásnak pénz és energia befektetését magyar banki rendszerek fejlesztésébe... Követésre méltó stratégia.

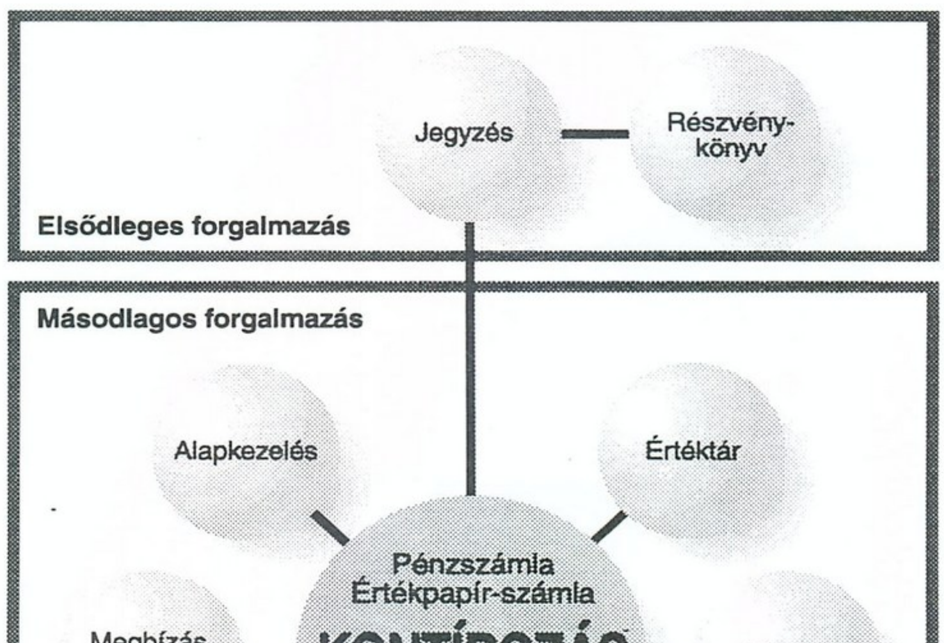
Multiuseres rendszerek

Immár hagyománnyá vált, hogy a HP az év elejére gyűjti össze egész éves üzletpolitikáját meghatározó termékbejelentéseit. Az idei újdonságok a többfelhasználós termékcsoportot érintették, amelyeknek mintegy 60%-a újult meg. Az X terminálok piacán egy alacsony árfekvésű, könnyen kezelhető családdal (Entria X) és egy X-terminálvezérlő szoftverrel (Enware) rukkolt ki a HP. Elsősorban olyan banki, pénzügyi, biztosítási és vevőszolgálati felhasználásokhoz ajánlják, ahol nagygépes környezetről térnek át hálózatos, Unix platformra, és a karakteres terminálokat lecserélik egy kliens/szerver környezetű grafikus állomásra. Az 1000 dollárnál kezdődő X terminálok a „plug and play" (azonnal csatlakoztatható) tulajdonság révén 10 percen belül rendszerbe állíthatók, üzembe helyezhetők. Az energiatakarékos, ergonomikus, ikonalapú képernyővel rendelkező X terminálok dinamikus billentyűzetleképzési funkciója lehetővé teszi a különböző billentyűzetkiosztást megkívánó alkalmazások ablakai közötti átkapcsolást.

Megújultak a munkaállomások is, két új típussal (712/60, 712/80i) gyarapodott a HP 9000 700-as sorozata. A PA-7100LC processzoron alapuló modellek multimédia-képességét továbbfejlesztették, és áruk is kedvező, olcsóbbak a — Magyarországon az induló kategóriában oly népszerű — Sun desktop-munkaállomásoknál. A 712-es modelleket a hagyományos felhasználási területek mellett (termékmenedzselés, vevőszolgálat, tőzsdei kereskedelem stb.) elsősorban a felgyorsított grafikai alkalmazásokra (CAD, CASE) optimalizálták. A rajzok, állóképek és videofelvételek színhű megjelenítésére képes modellek (8 millió színnel) a HP-UX 9.3 operációs rendszer alatt üzemelnek, és binárisan kompatibilisek a HP 9000 700-as sorozat munkaállomásaival.

Három új modellel (E25/48, E35/64, E45/80) bővítette szerverkínálatát a HP az „alsó végeken". A szervereket ügyes marketingfogással igyekeznek elfogadtatni a magyar vásárlókkal: felső kategóriás PC-áron (6500 — 113 000 $) kínálnak AS/400, VAX/VMS teljesítményt...

A hardverbejelentések természetesen szoftverújdonságokkal is párosultak. Ezek közül az egyik legfontosabb, hogy



elkészült a HP hálózati menedzsment szoftverének legújabb változata (OpenView 3.3). A legnagyobb sikerre azonban — itthon és külföldön — a Windows-emulátorok számíthatnak: a Wabi 1.1 és az Insignia SoftWindows.

Egy „felnőtt" alkalmazásfejlesztő

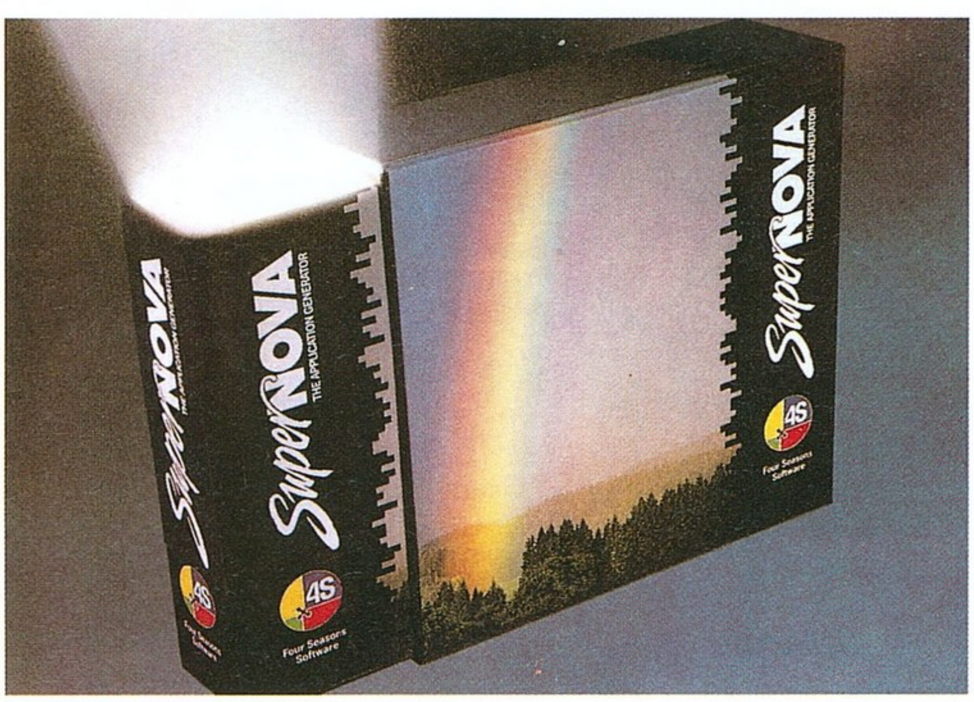
Napjainkban gombamódra szaporodnak a 4GL-nek hívott alkalmazásfejlesztő eszközök (Oracle, Ingres, Magic, Gupta, ...). A hazai piacon ezt a kínálatot bővíti a nemrég megjelent, korszerű fejlesztési technológiát képviselő termék, a SuperNOVA. A Four Seasons cég alkalmazásfejlesztő környezetének disztribúciós és support feladatait az — eddig elsősorban hardverorientált — Albacomp látja el (új, budapesti consulting csoportja segítségével).

A Unix, Vax/VMS, NOVELL, DOS, MS-Windows, NT platformokon elérhető SuperNOVA erős konkurenciát jelent az eddig már „befutott" alkalmazásfejlesztő eszközöknek; ezek közül egy fejlesztő — ha hosszú távon gondolkodik — olyat választ, amely kielégít egy sor követelményt.

Alapvető, hogy az alkalmazásfejlesztő támogassa a kliens/szerver architektúrát, objektumorientált legyen, és hatékony grafikus interaktív fejlesztési környezettel rendelkezzen (drag and drop stílus, áttekinthető menürendszer). Fontos, hogy az — akár háttér-adatbáziskezelő nélkül kifejlesztett — alkalmazás a későbbiekben tetszőleges adatbázisokon (SQL vagy más, például C-Isam, dBase, Btrieve) képes legyen futni. Lényeges az osztott feldolgozás és az osztott adatbáziskezelés transzparens támogatása. Szükséges, hogy az alkalmazás újrafordítás nélkül hordozható legyen különböző platformok és felhasználói interfészek között (például MS-Windows környezetben fejlesztett grafikus alkalmazás futtatása X terminálon, vagy egyszerű karakteres terminálon).

Ha e feltételek szerint válogatjuk az alkalmazásfejlesztők mezőnyét, akkor a versenyzők megfogyatkoznak. Az agresszív runtime árképzésű (leszűkített rendeltetésű) SuperNOVA — a disztribútor állítása szerint — azonban megfelel ezen feltételek mindegyikének.

Más jellegzetes vonás is megkülönbözteti a most debütáló SuperNOVA-t. Nem akarja pótolni a fejlesztőt, hanem csak segíteni annak munkáját. Az interaktív menüvezérelt lehetőségek mellett rendelkezik egy hatékony 4GL nyelvvel, amelyben leírható az alkalmazás logikája. Így a különböző módszerekkel készült alkalmazás teljes 4GL kódja előállítható, amely módszer jól elősegíti a verziókontrollt és a csoportmunkát.



A SuperNOVA másik markáns jellegzetessége az adatbázis-interfész és az osztott feldolgozás minőségére vonatkozik. Egy elkészült alkalmazás által használt adattáblák és eljárások (!) üzemeltetés közben úgy oszthatók szét a hálózaton, hogy magához az alkalmazáshoz nem kell hozzányúlni. Ez azt jelenti, hogy a kliensoldalon található (az erőforrások helyét megjelölő) szétosztási tábla egyszerű módosításával eldönthető, melyik adattáblát melyik szerveren tartsa a rendszer. Sőt, az alkalmazás tetszőleges eljárása egy távoli szerveren futhat le — például azért, mert az összes olyan adat ott van, amelyet az adott modul használ. Ezt az osztott feldolgozást a SuperNOVA egy hálózati protokollra (TCP/IP, X.25) épülve önmaga oldja meg, az adatbáziskezelő hálózati interfészének (SQL*NET, Ingres/NET, I-NET,...) felhasználása nélkül. A kommunikáció így sokkal hatékonyabb, paraméterezhetően tömörített vagy kódolt.

Precízen, pontosan, megbízhatóan

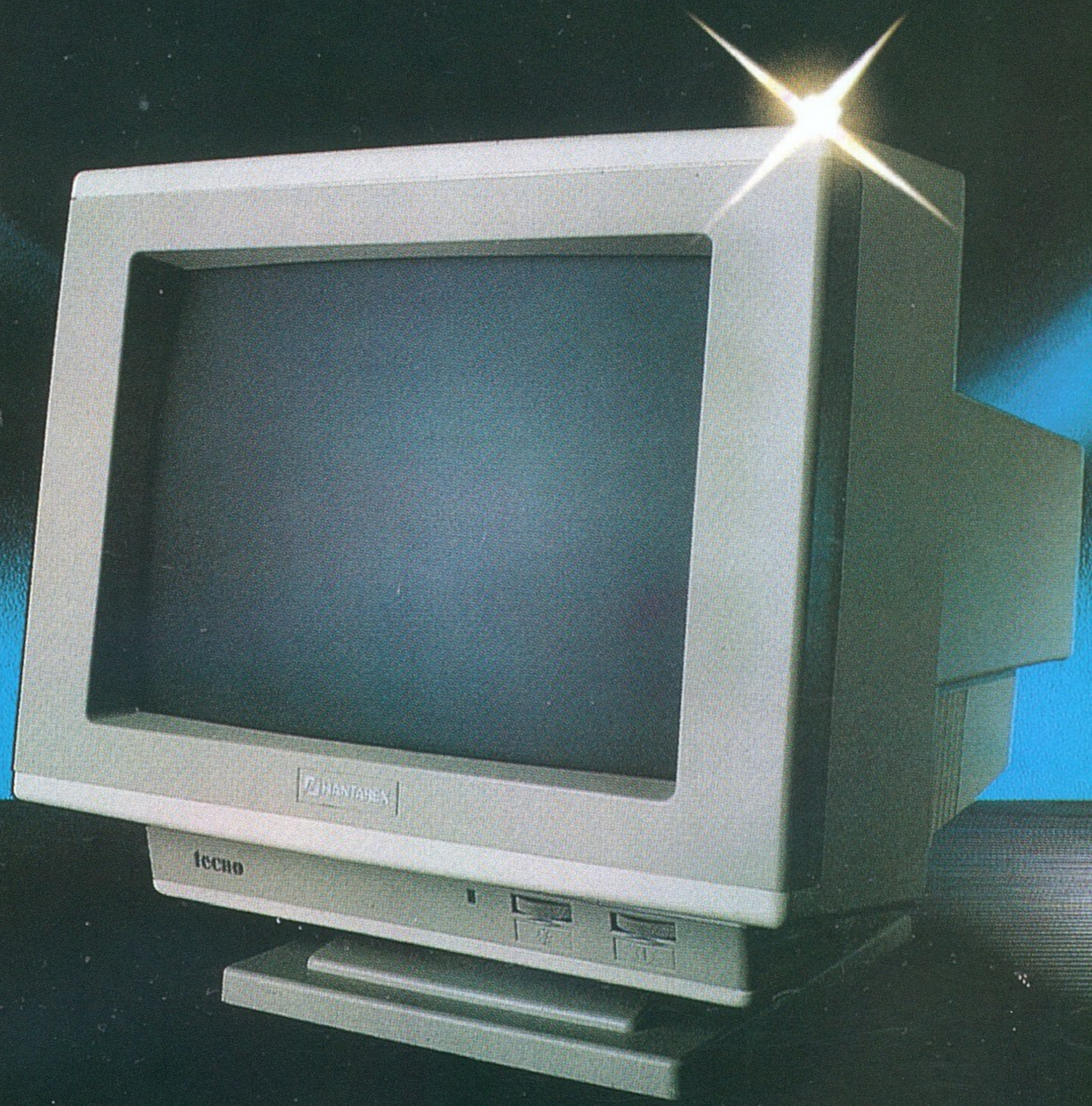
Aki adminisztratív munkakörben dolgozik, sokszor kétségbeesetten keresgél, amikor egy félbehagyott jelentést akar folytatni, vagy ha egy már lerakott iratba szeretne bepillantani. Nagy mennyiségű dokumentumok tárolására, és hatékony, gyors visszakeresésére már jó néhány szoftver (Doktár-Archiware, FileNet, PleXus stb.) rendelkezésre áll, ezek azonban általában sokkal többet tudnak, mint amire a felhasználók többségének szüksége van, nem beszélve e rendszerek magas áráról. Felismerték, hogy sokkal hamarabb vesz meg egy cég egy egymásra épülő, moduláris archiváló-visszakereső rendszert, amelyet később igényeinek és pénztárcájának megfelelően bővíthet. A Doktár — az egyik legnépszerűbb hazai archiváló/visszakereső rendszer — telepítései kapcsán a fejlesztő Hypermédia és a forgalmazó IQ SOFT ráérzett a felhasználók vásárlási szokásaira, s a Banktechen kirukkolt a Doktár „modulokra szedett" változatával.

A DA-családból (DA-Iktat, DA-Dosszié, DA-...) a legnagyobb sikerre valószínűleg a DA-Dosszié számíthat, hiszen 100 000 Ft alatti árával sok felhasználó számára elérhető. Az irodai adminisztrációt könnyítő windowsos keretrendszer összefogja a Word szövegszerkesztőt, és az Excel táblázatkezelővel vagy más alkalmazással készülő munkákat. A DA-Dossziéval a dokumentum automatikusan bekerül egy elektronikus dossziéba, amelynek fedőlapján a hagyományos papírdossziéhoz képest jóval több információ (leíró) adható meg. A tárolt állományok e leírók alapján visszakereshetők vagy áttehetők egy alkalmazásba. Lehetőség van arra, hogy különböző dossziéfedőlapokat alakítsunk ki, így egy dosszié többfajta alkalmazásra is felhasználható. A hatékony adatbázisszervezésnek köszönhetően kereshetünk a szöveges dokumentum bármely szava, vagy ezek kombinációja alapján is.

A DA-Dosszié jól használható mint elektronikus iktatókönyv, de sajtófigyelésre (cikkek archiválására, a cikkekben szereplő szavak szerinti visszakeresésre) és projektkövetésre (a projekttel kapcsolatos információk tárolására, és az ehhez tartozó részfeladatok és határidők követésére) is. A szoftver — szűkített adatbázis-változattal — fut notebookon, de építőkocka módszerrel bővíthető is, így nyílt rendszerként egyre nagyobb feladatokra használható, akár hálózatban is, több munkaállomáson.

Sziebig Andrea

MONITOR COLORE VGA





- Kiváló minőségű, hazai gyártású monitorok
- VGA–SVGA, 14'' méretben
- Alacsony sugárzású kivitelben is

Viszonteladóknak kedvezmény!

HANTAREX INDUSTRY RT.

Pécs, Szilva u. 1–3. Telefon: (72) 439-766, (72) 439-751

INFORMÁCIÓKÉRÉS: A0318 ▶

SuperNOVA
THE APPLICATION GENERATOR

4S
Four Seasons Software

Tisztelt Kolléga!

Ön már jó ideje gondolkodik azon, hogy milyen alkalmazásfejlesztő technológiával biztosíthatná csapatának eredményes munkáját. Ha hosszútávú megoldást keres, akkor biztosan egyetért a következő szempontokkal:

— Nagy hatékonyságú objektumorientált grafikus fejlesztő környezet
— Bővíthető 4GL nyelv
— Többszintű, jól definiált API-k
— Adatbázisfüggetlenség (SQL: Oracle, Ingres, DB2/6000, Informix, Sybase,... RDB, C-ISAM, dBase, Btrieve...)
— Osztott adatbáziskezelés
— Referenciális integritás és tranzakciókezelés, akár különböző adatbázisok között is
— Kliens-szerver architektúra
— Távoli eljáráshívások
— Platformfüggetlenség (UNIX, VAX/VMS, OS/400, NOVELL, MS-DOS, Windows, NT)
— Grafikus vagy karakteres megjelenítés, újrafordítás nélkül
— Többszintű jogosultságkezelés
— Verziókontroll és csoportmunka támogatása
— Kapcsolat CASE eszközökhöz
— Elképesztően jó runtime árkonstrukció

Ha megér Önnek egy napot, kérjük, vegyen részt ingyenes, tutorial jellegű bemutatónkon. Heterogén számítógépes és adatbáziskörnyezetben mutatjuk be Önnek a SuperNOVA fenti tulajdonságait.

Jelentkezés:

ALBACOMP
CONSULTING GROUP

Telefax: (36-1) 252-7926, (36-22) 327-532

INFORMÁCIÓKÉRÉS: A0301 ▲